Jun 10, 2024

# Análise de metabarcoding de DNA

DOI
https://dx.doi.org/10.17504/protocols.io.yxmvme9k5g3p/v1

Thiago Mafra Batista[1]

[1]Universidade Federal do Sul da Bahia

Thiago Mafra Batista
Universidade Federal do Sul da Bahia





**DOI:** https://dx.doi.org/10.17504/protocols.io.yxmvme9k5g3p/v1

**Protocol Citation:** Thiago Mafra Batista 2024. Análise de metabarcoding de DNA. **protocols.io** https://dx.doi.org/10.17504/protocols.io.yxmvme9k5g3p/v1



**Protocol status:** Working
**We use this protocol and it's working**

**Created:** June 09, 2024

**Last Modified:** June 10, 2024

**Protocol Integer ID:** 101460

**Keywords:** 16SV3V4, metabarcoding, bioinformatics, metagenomics, usearch, mapseq, realizarem análise metataxonômica de microrganismo, dna este tutorial conduzirá, operational taxonomic unit, gene 16s ribossomal, partir da amplificação das regiões v3v4, das sequência, dna, similaridade

## Abstract

Este tutorial conduzirá estudantes e pesquisadores a realizarem análise metataxonômica de microrganismos a partir da amplificação das regiões V3V4 do gene 16S ribossomal. Iniciando pela união dos pares de reads com ***usearch***, em seguida, usamos o ***cutadapt*** para cortar os primers das sequências. Depois, realizamos a filtragem de qualidade com *usearch*, encontramos sequências únicas e abundantes, e criamos clusters de OTUs (Operational Taxonomic Units) com 97% de similaridade. Em seguida, utilizamos o ***mapseq*** para mapear as OTUs contra um banco de dados. Por fim, criamos tabelas de OTUs para cada amostra e geramos um arquivo HTML para visualização dos resultados com o ***Krona***.

## Materials

Softwares necessários para toda pipeline:

- usearch (**https://www.drive5.com/usearch/**)
- cutadapt (https://cutadapt.readthedocs.io/en/stable/index.html)
- MAPSeq (https://github.com/jfmrod/MAPseq)
- Krona (https://github.com/marbl/Krona/wiki)
- R
- awk

Os scripts acessórios estão com link do github disponível no tutorial.

## Análise da qualidade das reads

1 Vamos avaliar o perfil de qualidade das reads com o software FastQC.

```
$ fastq -t 64 *.fastq -o .
```

## União dos pares de reads

2 Cada par de leitura é oriunda de um fragmento de amplicon sequenciado. Este fragmento precisa ser reconstruído. O parâmetro *-fastq_mergepairs* do *usearch* faz isso. Troque "{AMOSTRAS}" por um código único de sua escolha.

```
$ usearch -fastq_mergepairs *1.fq -reverse *2.fq -fastq_maxdiffs 6
-fastqout {AMOSTRAS}_merged.fq -relabel @ 2>>
{AMOSTRAS}_merged.log
```

## Remoção dos primers

3 As sequências dos primers também são sequenciadas e precisamos removê-las da reads, agora, unidas. Vamos utilizar o cutadapt e as sequências dos primers forward (parâmetro -g) e reverse (parâmetro -a).

```
$ cutadapt -g TCGTCGGCAGCGTCAGATGTGTATAAGAGACAGCCTACGGGNGGCWGCAG
{AMOSTRA}_merged.fq --discard-untrimmed 2>>{AMOSTRA}_cut.log |
cutadapt -a
GTCTCGTGGGCTCGGAGATGTGTATAAGAGACAGGACTACHVGGGTATCTAATCC -o
{AMOSTRA}_merged_cut.fq --discard-untrimmed - 1>>{AMOSTRA}_cut.log
```

## Descarte de reads com erros

4 Vamos agora descartar reads que provavelmente possuem erros, tanto nas sequencias que tiveram primers removidos, quanto nas sequencias sem primers identificados.

```
$ usearch -fastq_filter {AMOSTRA}_merged_cut.fq -fastq_maxee 1.0 -
fastaout {AMOSTRA}_filtered.fa -relabel Filt

$ usearch -fastq_filter {AMOSTRA}_merged.fq -fastq_maxee 1.0 -
fastaout {AMOSTRA}_filtered.fa -relabel Filt 2>>
{AMOSTRA}_filter.log
```

## Sequências únicas e abundantes

5 Agora vamos encontrar as sequências únicas e abundantes com o parâmetro -*fastx_uniques*

```
$ usearch -fastx_uniques {AMOSTRA}_filtered.fa -fastaout
{AMOSTRA}_uniques.fa -sizeout -relabel Uniq 2>>
{AMOSTRA}_uniques.log
```

## Clusters de OTUs

6 Agora vamos agrupar as sequências únicas e abundantes de acordo com a identidade entre elas, considerando 97% de similaridade.

```
$ usearch -cluster_otus {AMOSTRA}_uniques.fa -relabel
{AMOSTRA}_OTU -otus {AMOSTRA}_otus.fa 2>>{AMOSTRA}_otus.log
```

## Tabelas de OTUs

7 Agora vamos criar uma tabela contendo as informações de OTUs:

```
$ usearch -otutab {AMOSTRA}_merged.fq -otus {AMOSTRA}_otus.fa -
strand plus -otutabout {AMOSTRA}_otutab.txt 2>>
{AMOSTRA}_otutab.log
```

## Plotar a curva de rarefação da diversidade alfa

8 Vamos analisar a diversidade alfa a partir da curva de rarefação. Para isso vamos utilizar o script ***rare.R***.

```
$ Rscript rare.R ${AMOSTRA}_otutab.txt
```

## Mapear as OTUs contra um banco de dados

9 Com o *mapseq* vamos fazer blast das OTUs contra um banco de dados. No exemplo a seguir, utilizamos 24 processadores

```
$ mapseq -nthreads 24 {AMOSTRA}_otus.fa >{AMOSTRA}_otus.mapseq 2>>
{AMOSTRA}_otus.mapseq.log
```

## Criar tabelas de OTUs para cada amostra

10 Vamos criar uma tabela que irá conter as informações do mapeamento de cada amostra a partir do resultado do *mapseq*. Vamos usar um *loop* para otimizar a execução do *awk*.

```
for i in {2..13}
do
  awk -F"\t" '{print $1,"\t",$'"$i"'}' {AMOSTRA}_otutab.txt | sed
's/ //g' > "C${i}G$(($i+1))_otutab.txt"
done
```

## Concatenar as tabelas geradas com usearch e mapseq

11 Vamos unir as tabelas geradas pelo usearch e pelo mapseq para serem visualizadas no krona. Vamos utilizar o script em perl **concat_OTUs_Taxons.pl**.

```
for i in {2..13}
do
  concat_OTUs_Taxons.pl "C${i}G$(($i+1))_otutab.txt"
{AMOSTRA}_otus.mapseq > "C${i}G$(($i+1))_results.txt"
done
```

12

```
$ ktImportText C*_results.txt -o ${AMOSTRA}_results.html
```

STEP CASE

## Resultado da análise de metabarcoding de DNA 1 step

Este é um exemplo de resultado gerado por essa análise. Cada amostra está representada em um gráfico de pizza com seus respectivos percentuais de abundância. Os 7 níveis de classificação taxonômica são mostrados.


16SV34_results.html
628KB

## Script para otimização da análise

13

```
#!/bin/bash


# Este script realiza uma análise metataxonômica da região 16S
V3V4

echo "\nAnálise metataxonômica 16S regiões V3V4 Iniciada"

# Variáveis e parâmetros
AMOSTRA="16SV34"
usearch="usearch" # Especificar caminho completo se necessário
primer5="TCGTCGGCAGCGTCAGATGTGTATAAGAGACAGCCTACGGGNGGCWGCAG" #
Primer forward
primer3="GTCTCGTGGGCTCGGAGATGTGTATAAGAGACAGGACTACHVGGGTATCTAATCC"
# Primer reverse

# Unir pares
echo "\nUNINDO OS PARES"
$usearch -fastq_mergepairs *1.fq -reverse *2.fq -fastq_maxdiffs 6
-fastqout ${AMOSTRA}_merged.fq -relabel @ 2>>
${AMOSTRA}_merged.log
echo "UNIÃO CONCLUÍDA"

# Remover primers
echo "\nCUTADAPT"
# Cortando os primers e mantendo apenas as reads que foram
cortadas
cutadapt -g "$primer5" ${AMOSTRA}_merged.fq --discard-untrimmed
2>>${AMOSTRA}_cut.log | cutadapt -a "$primer3" -o
${AMOSTRA}_merged_cut.fq --discard-untrimmed -
1>>${AMOSTRA}_cut.log
echo "CUTADAPT CONCLUÍDO"

# Descartar reads com erros
echo "\nDESCARTANDO READS COM ERROS"
# Quality filtering: descartar reads que provavelmente possuem
erros
$usearch -fastq_filter ${AMOSTRA}_merged_cut.fq -fastq_maxee 1.0 -
fastaout ${AMOSTRA}_filtered.fa -relabel Filt
2>>${AMOSTRA}_filter.log
echo "DESCARTE CONCLUÍDO"

# Encontrar sequências únicas e abundantes
echo "\nENCONTRAR ÚNICAS E ABUNDANTES"
```

```
$usearch -fastx_uniques ${AMOSTRA}_filtered.fa -fastaout
${AMOSTRA}_uniques.fa -sizeout -relabel Uniq
2>>${AMOSTRA}_uniques.log

# Criar clusters de OTUs com 97% de similaridade
echo "\nCRIAR CLUSTERS DE OTUs COM 97% DE SIMILARIDADE"
$usearch -cluster_otus ${AMOSTRA}_uniques.fa -relabel
${AMOSTRA}_OTU -otus ${AMOSTRA}_otus.fa 2>>${AMOSTRA}_otus.log

# Criar tabelas de OTUs
echo "\nCRIAR TABELAS COM OS OTUs"
$usearch -otutab ${AMOSTRA}_merged.fq -otus ${AMOSTRA}_otus.fa -
strand plus -otutabout ${AMOSTRA}_otutab.txt
2>>${AMOSTRA}_otutab.log

# Criar e plotar a tabela de diversidade alfa (opcional)
#echo "\nCRIAR E PLOTAR A TABELA DE DIVERSIDADE ALFA"
#Rscript /home/ufsb/bin/rare.R ${AMOSTRA}_otutab.txt

# Mapear as OTUs contra banco de dados
echo "\nMAPEAR AS OTUs CONTRA BANCO DE DADOS"
mapseq -nthreads 8 ${AMOSTRA}_otus.fa > ${AMOSTRA}_otus.mapseq
2>>${AMOSTRA}_otus.mapseq.log

# Criar a tabela de OTUs para cada amostra
echo "\nCRIAR A TABELA DE OTUs PARA CADA AMOSTRA"
for i in {2..13}
do
  awk -F"\t" '{print $1,"\t",$'"$i"'}' ${AMOSTRA}_otutab.txt | sed
's/ //g' > "C${i}G$(($i+1))_otutab.txt"
done

# Unir as tabelas geradas pelo usearch e mapseq para abrir no
Krona
echo "\nUNIR AS TABELAS GERADAS PELO USEARCH E MAPSEQ PARA ABRIR
NO KRONA"
for i in {2..13}
do
  concat_OTUs_Taxons.pl "C${i}G$(($i+1))_otutab.txt"
${AMOSTRA}_otus.mapseq > "C${i}G$(($i+1))_results.txt"
done

# Criar um arquivo HTML para visualização dos resultados
echo "\nCRIAR UM ARQUIVO HTML PARA VISUALIZAÇÃO DOS RESULTADOS"
ktImportText C*_results.txt -o ${AMOSTRA}_results.html
```

```
echo "\nANÁLISE 16SV3V4 FINALIZADA""▫
```