May 28, 2024

# Integrated Bioinformatics Approach to Metabolite Detection from Metagenomic Data

DOI

dx.doi.org/10.17504/protocols.io.kxygxy35dl8j/v1

Vidya Niranjan[1], Lavanya C[1], Pooja Hoovina Venkatesh[1], Karthik.V[1], Sahana R[1]

[1] R V College of Engineering

Centre of Excellence in ...

Vidya Niranjan

R V College of Engineering

**DOI:** https://dx.doi.org/10.17504/protocols.io.kxygxy35dl8j/v1

**Protocol Citation:** Vidya Niranjan, Lavanya C, Pooja Hoovina Venkatesh, Karthik.V, Sahana R 2024. Integrated Bioinformatics Approach to Metabolite Detection from Metagenomic Data. **protocols.io** **https://dx.doi.org/10.17504/protocols.io.kxygxy35dl8j/v1**

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

**Created:** May 24, 2024

**Last Modified:** May 28, 2024

**Protocol Integer ID:** 100508

**Keywords:** Metagenome, Metabolites, Antismash, BLAST, integrated bioinformatics approach to metabolite detection, identifying biosynthetic gene cluster, general for the metabolite identification, identification of metabolite, identifying metabolite, complex nature of metagenomic dataset, metabolite identification, metagenomic dataset, metabolite detection, biosynthetic gene cluster, metagenomic data the exploration, metagenomic data, metagenome data, advanced bioinformatics tool, metabolite, integrated bioinformatics approach, biochemical diversity, presence of unknown enzyme, unknown enzyme, biotechnology, novel bioactive compound, pharmaceutical, bacterial species

## Abstract

The exploration of metabolites derived from metagenomic data holds immense potential for the discovery of novel bioactive compounds with applications in pharmaceuticals, agriculture, and biotechnology. However, the complex nature of metagenomic datasets, coupled with the biochemical diversity and the presence of unknown enzymes, poses significant challenges to identifying metabolites accurately. This study presents the development of a comprehensive protocol designed to streamline the identification of metabolites from metagenome data. The protocol integrates advanced bioinformatics tools and databases, including antiSMASH for identifying biosynthetic gene clusters (BGCs).
This study involves the identification of metabolites from bacterial species which can be potential Biofertilizers. But the pipeline can be used in general for the metabolite identification for any case study.

## Guidelines

The Current protocol was run in UBUNTU 20.04. But the protocol can be run in any version of UBUNTU. Tools have to be installed accoringly.

## Troubleshooting

## Retrieval of 16s Metagenomic Data

1 The Mulberry Rhizosphere 16s metagenomic samples were collected from NCBI SRA. The Illumina MiSeq platform was utilized for metagenomic analysis using an amplicon sequencing strategy. The source material consisted of environmental or biological samples from which DNA was extracted. The target regions of interest within the metagenome were selectively amplified using polymerase chain reaction (PCR). This amplification step was crucial for enriching specific genetic markers or regions, facilitating a detailed examination of the microbial community composition and diversity. The sequencing was performed in a paired-end layout, which involves reading both ends of each DNA fragment. This approach enhances the accuracy of sequence data and allows for better reconstruction of the amplicons. The resulting data provide insights into the genetic makeup and variability of the microbial populations within the sampled environment, enabling comprehensive metagenomic studies.

Dataset

16s metagenome data of Mulberry rhizosphere samples NAME

https://www.ncbi.nlm.nih.gov/sra/?term=PRJNA909945 LINK

## Taxonomical classification using USearch tool

2 USEARCH-tool was performed to obtain the taxonomical classification and OTU clustering table. The following is the USEARCH algorithm: USEARCH reads a file containing DNA sequences and sorts them in non-increasing length order. The first step is to combine the fastq files with all of the reads to determine which sample each read belongs to, USEARCH enables us to relabel the reads by prepending the sample name, allowing us to determine which read belongs to which sequence. After that, all combined files are merged to form the merged fasta file. To obtain high-quality read sequences, the reads are trimmed and filtered, yielding the filtered fasta file. De-replication is used to avoid analysing the same sequencetwice. De-replication locates a set of distinct sequences in an input file. Sequences are compared letter by letter and must be identical along their entire length

(substrings do not match because case is ignored, an upper-case letter will match a lower-case letter). The sintax
command predicts taxonomy for query sequences in FASTA or FASTQ format using the SINTAX algorithm. Taxonomy assignment is accomplished by comparing our sequences to
the many databases that are available.

2.1 The first step in Usearch is merging the fastq files.

Command

### Merging Files

Merging Files

```
usearch -fastq_mergepairs "E:\SRR_1.fastq" -reverse "E:\SRR_2.fastq" -
fastqout merged_file.fastq
```

2.2 Filtering the merged reads

Command

### Filteration

Filteration

```
usearch -fastq_filter merged_file.fastq -fastq_maxee 1.0 -
fastqout file_fil.fasta
```

```
fastq_maxee
```

The parameter is used in the context of filtering and quality control of sequencing data, particularly in software tools designed to process FASTQ files. FASTQ files contain both the sequence data and the corresponding quality scores for each base.
The

```
fastq_maxee
```

The paramete is used in the context of filtering and quality control of sequencing data, particularly in software tools designed to process FASTQ files. FASTQ files contain both the sequence data and the corresponding quality scores for each base.
The , which reflect the probability of an incorrect base call. The formula to calculate the expected errors for a read is the sum of the error probabilities for each base in the read.
For example, if a read has a

```
fastq_maxee
```

of 1.0, it means that the total expected number of errors for that read should not exceed 1.0. Reads with expected errors higher than this threshold are discarded from further analysis. This filtering step helps ensure that the remaining reads are of high quality, reducing the likelihood of introducing erroneous sequences into the downstream analysis.
By setting

```
fastq_maxee
```

to 1.0, researchers can balance between retaining a sufficient number of reads and ensuring high data quality. Lowering the threshold (e.g., to 0.5) would result in stricter filtering, while increasing it (e.g., to 2.0) would be more lenient, potentially allowing more low-quality reads to pass through.

2.3 usearch: This invokes the USEARCH program, a popular software for processing and analyzing high-throughput sequencing data.
-fastx_uniques file_fil.fasta: This option tells USEARCH to process the input FASTA file named file_fil.fasta. The -fastx_uniques command is used to find unique sequences in the input file. It collapses identical sequences into unique sequences and counts the number of occurrences of each sequence.
-fastaout file_u.fasta: This specifies the name of the output FASTA file, file_u.fasta, where the unique sequences will be saved. Each unique sequence will be written to this file.
-relabel uniq: This option relabels the sequences in the output file. Each unique sequence will be given a label starting with “uniq” followed by a number (e.g., uniq1, uniq2, etc.). This makes it easier to identify and reference each unique sequence.

-sizeout: This option appends the size (i.e., the number of times each unique sequence was found in the original input file) to the label of each sequence in the output file. The format will be ;size=N where N is the count of that sequence in the input data. This provides additional information about the abundance of each unique sequence.

Command

### Finding the unique reads

Finding the unique reads

```
usearch -fastx_uniques file_fil.fasta -fastaout file_u.fasta -relabel
uniq -sizeout
```

2.4 usearch: This invokes the USEARCH program, a versatile tool for high-throughput sequencing data analysis.
-makeudb_usearch rdp_16s_v16.fa: This option specifies that a USEARCH database should be created from the input FASTA file named rdp_16s_v16.fa. The input file typically contains 16S rRNA gene sequences, which are commonly used for identifying and classifying bacteria and archaea.
-output rdp_16s.udb: This option specifies the name of the output file where the USEARCH database will be saved. The output file, rdp_16s.udb, will be in a binary format specific to USEARCH, optimized for fast searching and alignment operations.

Command

### Make Database and indexing

### Make Database and indexing

```
usearch -makeudb_usearch rdp_16s_v16.fa -output rdp_16s.udb
```

2.5 usearch: This invokes the USEARCH program, a versatile tool for high-throughput sequencing data analysis.
-sintax merged_file.fastq: This option specifies that the SINTAX algorithm should be used to perform taxonomic classification on the sequences in the input FASTQ file named merged_file.fastq. The SINTAX algorithm assigns taxonomy to sequences based on a reference database.
-db rdp_16s.udb: This specifies the path to the USEARCH-formatted reference database (rdp_16s.udb) that was previously created. This database contains 16S rRNA gene sequences and their associated taxonomy, which will be used for classifying the input sequences.
-tabbedout file.sintax: This option specifies the name of the output file (file.sintax). The results of the taxonomic classification will be saved in this file in a tab-separated format. Each line in the output file will correspond to a sequence from the input file and will include information about its taxonomic classification and confidence scores.
-strand both: This option indicates that both strands of the DNA sequences should be considered during the classification process. This means that the algorithm will check both the forward and reverse complements of the sequences to find the best match in the reference database.
-sintax_cutoff 0.8: This option sets the confidence cutoff for taxonomic assignments. A SINTAX score (confidence value) of at least 0.8 is required for a taxonomic assignment to be accepted. Scores range from 0 to 1, with higher scores indicating higher confidence in the classification.

Command

### Taxonomical classification

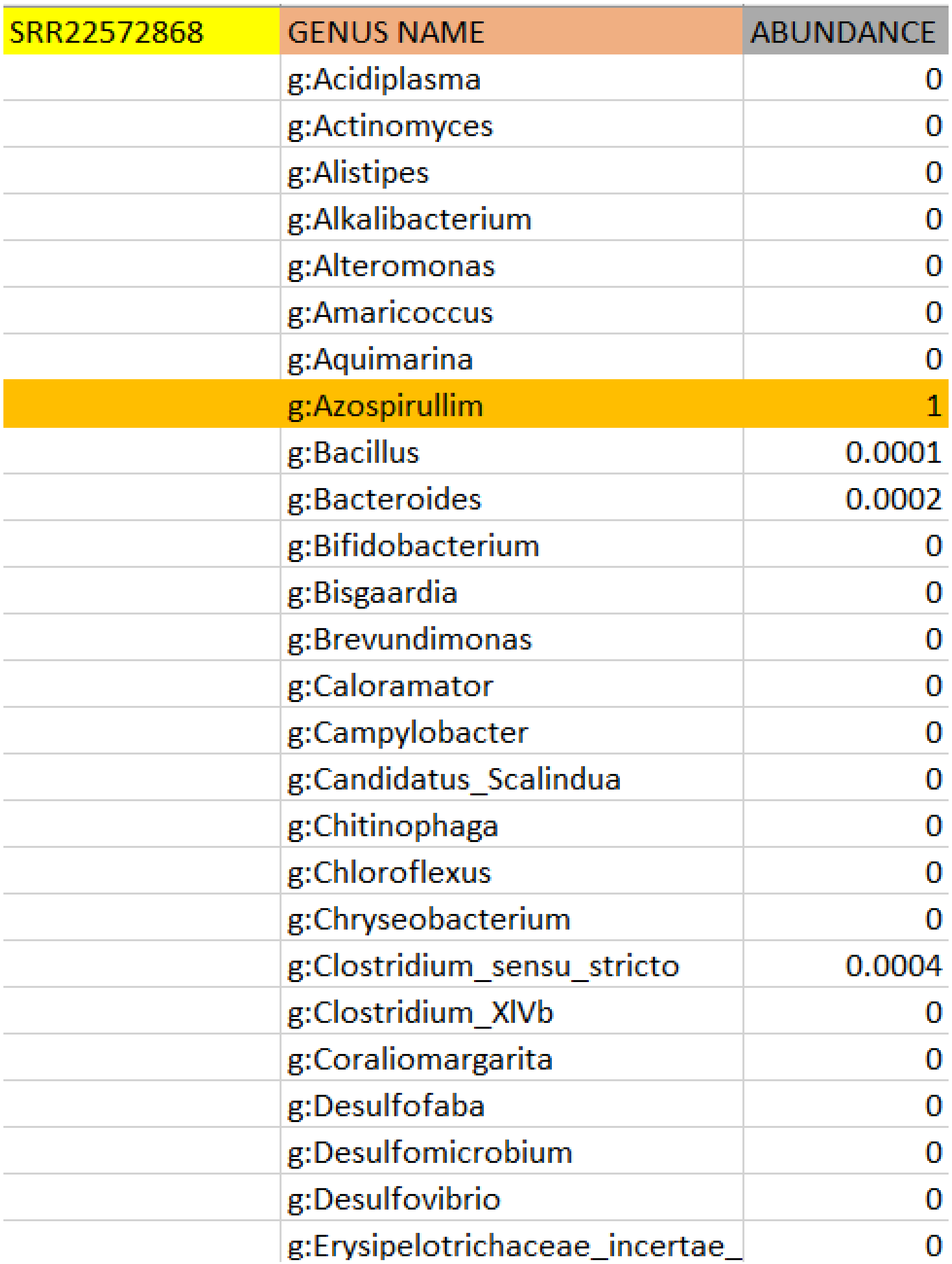
Taxonomical classification

```
usearch -sintax merged_file.fastq -db rdp_16s.udb -tabbedout
file.sintax -strand both -sintax_cutoff 0.8
```

The sintax cutoff varies for different samples. So it should be set according to the samples chosen.

2.6 The commands mentioned above were run in windows 11 after installation of the latest version of USearch (v11). The scripts were saved as a batch file (.bat) and was automated.

Expected result



| SRR22572868 | GENUS NAME | ABUNDANCE |
|---|---|---|
| | g:Acidiplasma | 0 |
| | g:Actinomyces | 0 |
| | g:Alistipes | 0 |
| | g:Alkalibacterium | 0 |
| | g:Alteromonas | 0 |
| | g:Amaricoccus | 0 |
| | g:Aquimarina | 0 |
| | g:Azospirullim | 1 |
| | g:Bacillus | 0.0001 |
| | g:Bacteroides | 0.0002 |
| | g:Bifidobacterium | 0 |
| | g:Bisgaardia | 0 |
| | g:Brevundimonas | 0 |
| | g:Caloramator | 0 |
| | g:Campylobacter | 0 |
| | g:Candidatus_Scalindua | 0 |
| | g:Chitinophaga | 0 |
| | g:Chloroflexus | 0 |
| | g:Chryseobacterium | 0 |
| | g:Clostridium_sensu_stricto | 0.0004 |
| | g:Clostridium_XlVb | 0 |
| | g:Coraliomargarita | 0 |
| | g:Desulfofaba | 0 |
| | g:Desulfomicrobium | 0 |
| | g:Desulfovibrio | 0 |
| | g:Erysipelotrichaceae_incertae_ | 0 |

Taxonomical classification from USearch

## Identification of the Metabolites from Metagenome data

3 Collection of the reference genomes

From the previous step, the users have to choose the specific genus with good abundance and collect the respective reference genomes from the NCBI REFSEQ portal. In this case, we have targeted specific species *Pseudomonas putida, Frankia casuarinae, Azospirullum brasilense, Frankia torreyi and*
*Frankia alni were chosen for identifying the metabolites associated with the microbes.*

Dataset

**Reference genomes from REFSEQ** NAME

https://www.ncbi.nlm.nih.gov/refseq/ LINK

3.1 Indexing and building the database

Latest version of BLAST + was installed and the retrieved sequences was indexed and species wise database was created with the MakeBlastDB option of BLAST+. In the Input option the user should give the specific organism reference sequence.

Command

**MakeBlastDB**

**MakeBlastDB**

```
makeblastdb -in reference_sequences.fasta -dbtype nucl -parse_seqids -
out my_database
```

3.2 Finding the Regions of Similarity

The 16s metagenome samples were subjected to BLASTn analysis with the database created
in the previous step, with this specific gene segment of the sample getting mapped to

the
reference genome with the identity percent was revealed with the analysis.

1. blastn: This specifies that you are using the BLASTN program, which is used for nucleotide sequence comparisons.
2. -query query_sequences.fasta: This option specifies the input query file in FASTA format that contains the nucleotide sequences you want to search against the database.
3. -db my_database: This specifies the BLAST database against which the query sequences will be searched. The database should have been previously formatted with makeblastdb.
4. -out results.out: This specifies the output file where the BLAST results will be saved.
5. -outfmt “6 qseqid sseqid pident length mismatch gapopen qstart qend sstart send evalue bitscore”: This option specifies the format of the BLAST output. In this case, 6 specifies tabular output without comments, and the following fields are included in the output:

qseqid: Query sequence ID
sseqid: Subject (database) sequence ID
pident: Percentage of identical matches
length: Alignment length (number of bases aligned)
mismatch: Number of mismatches
gapopen: Number of gap openings
qstart: Start of alignment in the query
qend: End of alignment in the query
sstart: Start of alignment in the subject
send: End of alignment in the subject
evalue: Expectation value (E-value) indicating the number of hits expected by chance
bitscore: Bit score of the alignment

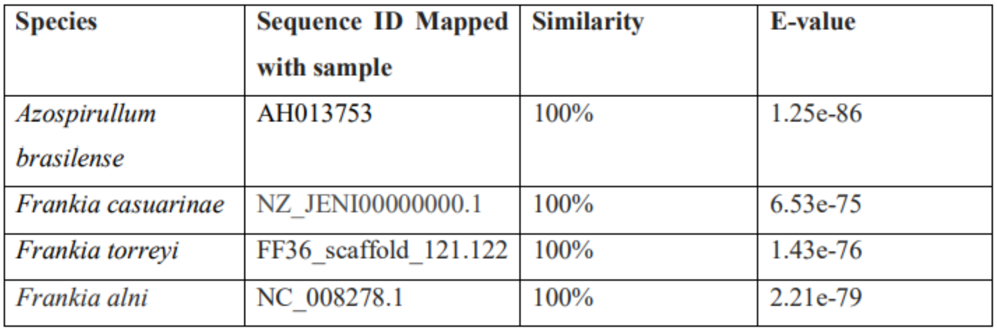
Command

Regions of similarity in BLAST

Regions of similarity in BLAST

```
blastn -query query_sequences.fasta -db my_database -out results.out -
outfmt "6 qseqid sseqid pident length mismatch gapopen qstart qend
sstart send evalue bitscore"
```

Expected result



| Species | Sequence ID Mapped with sample | Similarity | E-value |
|---|---|---|---|
| *Azospirullum brasilense* | AH013753 | 100% | 1.25e-86 |
| *Frankia casuarinae* | NZ_JENI00000000.1 | 100% | 6.53e-75 |
| *Frankia torreyi* | FF36_scaffold_121.122 | 100% | 1.43e-76 |
| *Frankia alni* | NC_008278.1 | 100% | 2.21e-79 |

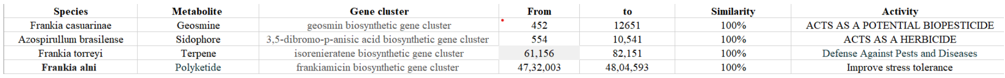
Similarity search from BLAST+

## Identification of Metabolites

4 The sequence ID of the Gene Segment was retrieved using GENBANK and it was subjected
to metabolite identification using ANTISMASH.
Exploring the secondary metabolism of bacteria and fungi presents significant opportunities for discovering new bioactive compounds, which can be valuable in

pharmaceuticals such as antibiotics, anti-tumor agents, and cholesterol-lowering drugs. However, identifying gene clusters responsible for secondary metabolite production in newly sequenced microbial genomes is a complex task. This complexity arises from the biochemical diversity, the presence of unknown enzymes, and the dispersed nature of bioinformatics tools. AntiSMASH (antibiotics & Secondary Metabolite Analysis Shell) offers a comprehensive solution to this challenge. It is capable of identifying biosynthetic loci for various classes of secondary metabolites, aligning these regions with known gene clusters, and integrating multiple secondary metabolite analysis methods into a single, user-friendly interface. By streamlining the identification of potential drug candidates, antiSMASH facilitates further research into microbial secondary metabolism. Using this tool, metabolites were identified from metagenomic data, enhancing the efficiency of discovering promising new metabolites.

Expected result



| Species | Metabolite | Gene cluster | From | to | Similarity | Activity |
|---|---|---|---|---|---|---|
| Frankia casuarinae | Geosmine | geosmin biosynthetic gene cluster | 452 | 12651 | 100% | ACTS AS A POTENTIAL BIOPESTICIDE |
| Azospirullum brasilense | Sidophore | 3,5-dibromo-p-anisic acid biosynthetic gene cluster | 554 | 10,541 | 100% | ACTS AS A HERBICIDE |
| Frankia torreyi | Terpene | isorenieratene biosynthetic gene cluster | 61,156 | 82,151 | 100% | Defense Against Pests and Diseases |
| **Frankia alni** | Polyketide | frankiamicin biosynthetic gene cluster | 47,32,003 | 48,04,593 | 100% | Improve stress tolerance |

Metabolites from Metagenome data