DISCo-microbe

Release 0.1

Dana L. Carper, Travis J. Lawrence, Alyssa A. Carrell, Dale A. Pell

Dec 05, 2019

CONTENTS:

1 Introduction **1**

2 DISCo-microbe Installation **3**
- 2.1 pip . . . 3
- 2.2 git . . . 3
- 2.3 anaconda . . . 3
- 2.4 Test your installation . . . 4

3 DISCo-microbe Tutorial **5**
- 3.1 Getting the data . . . 5
- 3.2 Create Module . . . 5
  - 3.2.1 Option to include specific strains . . . 6
  - 3.2.2 Option to start with input database . . . 6
- 3.3 Subsample Module . . . 6
  - 3.3.1 Subsample by total number of members to include . . . 6
  - 3.3.2 Subsample by proportions . . . 7
- 3.4 18S example . . . 7
- 3.5 Tutorial Completed . . . 7

4 File Format Descriptions **9**
- 4.1 Create module . . . 9
- 4.2 Subsample module . . . 10

5 Indices and tables **13**

CHAPTER

ONE

INTRODUCTION

Design of an Identifiable Synthetic Community of Microbes (DISCo-microbe) is an easy-to-use command-line program, for creation of diverse communities of organisms that can be distinguished through next-generation sequencing technology. DISCo-microbe consists of two modules, create and subsample. The create module constructs a highly diverse community at a specified sequence difference from an input of aligned DNA/RNA sequences, e.g., 16S sequence. The module can either design a de novo community or design a community that includes targeted organisms. create solves problem (1) by easily generating a diverse community of members through an easily documentable method, ensuring reproducibility. The subsample module provides options for dividing the community into subsets, according to either the number of members or the proportions of a grouping variable, both of which can be specified by the user. subsample module solves problem (2) by allowing the user to subsample an already distinguishable community of members based on attributes of interest. Although this software was designed for construction of microbial communities, any DNA/RNA alignment can be used as input; consequently, users are not restricted to any particular organismal group or marker gene. This program is implemented in Python and is available through GitHub and PYPI.

CHAPTER

TWO

DISCO-MICROBE INSTALLATION

DISCo-microbe is a python based package that can easily be installed with `pip` or directly from our github page.

2.1 pip

If you have a native installation of `python` you can use:

```
>>> pip install disco-microbe
```

2.2 git

The newest development version of disco-microbe may be installed from our github page. You can use git directly, or download a zipfile.

```
>>> git clone https://github.com/dlcarper/Disco-microbe.git
>>> cd Disco-microbe
>>> python setup.py install
```

2.3 anaconda

Anaconda provides the `conda` environment and is the recommended way to install DISCO-microbe if you are operating a Windows machine without a native `python`. Once you have `anaconda` installed, create a `conda` environment and then use `source` and `activate` to launch a `python` environment.

```
>>> conda create -n disco-microbe-env python=3.7
>>> source activate disco-microbe-env
```

You can now proceed with a pip install:

```
>>> pip install disco-microbe
```

2.4 Test your installation

You can test your installation by running the DISCo-microbe help command:

```
>>> disco -h
```

CHAPTER

THREE

DISCO-MICROBE TUTORIAL

This tutorial covers a DISCo workflow to generate a list of organisms that are identifiable at a user-specified edit distance (or minimum number of nucleotide differences) for a targeted region of DNA. The goal is to become familiarized with the general workflow, data files, and parameter settings in DISCo. We will use a referenced-based alignment of 16S rRNA gene sequences trimmed to the V4 region as an example, but the workflow applies to any other marker gene or organisms (e.g. fungal or other organism). Follow along by copy/pasting the code-blocks into a command line terminal.

Note If you haven't installed DISCo yet, go here first: Installation

3.1 Getting the data

Use the commands below to download and extract the data in a command-line interface. This will create a new directory called `TUTORIAL_FILES/` located in your current directory.

```
>>> wget -L https://github.com/dlcarper/DISCo-microbe/raw/master/TUTORIAL_FILES.tar.gz
>>> tar -xvzf TUTORIAL_FILES.tar.gz
```

Use the command `cd` to navigate and `ls` to look inside the `TUTORIAL_FILES/` directory. You will see that it contains different files to input in each module.

```
>>> cd TUTORIAL_FILES/
>>> ls
```

```
18S_example/
RDP_distance_dictionary_20191126-150855.txt
RDP_Tutorial_Metdata_file.txt
RDP_Tutorial_alignment.fasta
RDP_Tutorial_proportions_file.txt
RDP_Tutorial_starter_community_file.txt
```

3.2 Create Module

The create module has two required arguments, an alignment of DNA or RNA sequences in FASTA format (`--i-alignment`) and a user-specified minimum sequence distance between community members (`--p-editdistance`). For this tutorial we will use an alignment of 16S rRNA genes that were subsampled from RDP, aligned with SINA, and columns with only gaps were removed. We use an edit distance of 3 so that our final community list will contain community members that have a minimum of 3 nucleotide differences. We will also include a seed for reproducibility purposes (`--p-seed`). To have taxonomic information in the final community list,

we also input a tab-delimited metadata file (`--i-metadata`) that contains the sequence identifier followed by the taxonomic identification for each sequence.

```
>>> disco create --i-alignment RDP_Tutorial_alignment.fasta --p-editdistance 3 --p-
↪seed 10 --i-metadata RDP_Tutorial_Metdata_file.txt --o-community-list community_ED3_
↪with_taxonomy.txt
```

This command should generate two files: a text file containing a list of members that differ by at least 3 nucleotides and a distance dictionary. The distance dictionary is a database of the pairwise sequence similarities for the provided community alignment. This distance database can be used as a starting point for the create module if you were to add community members to your existing alignment and only needed to calculate the pairwise distances for the new members.

3.2.1 Option to include specific strains

If you wish to generate a community that includes specific strains, you may add an additional argument (`--p-include-strains`) of a list of strains that must be included in the final community.

> **Note** The list of strains must have an edit distance greater than or equal to the edit distance you provide for the final community. If your input list of strains have less nucleotide differences than the edit distance you specify, you will receive a conflict warning and must address the conflicts before proceeding.

```
>>> disco create --i-alignment RDP_Tutorial_alignment.fasta --p-editdistance 3 --p-
↪seed 10 --i-metadata RDP_Tutorial_Metdata_file.txt --o-community-list community_ED3_
↪with_taxonomy_specific.txt  --p-include-strains RDP_Tutorial_starter_community_file.
↪txt
```

3.2.2 Option to start with input database

If you wish to start with a pre-existing distance database simply specify in your create command

```
>>> disco create --i-alignment RDP_Tutorial_alignment.fasta --p-editdistance 3 --p-
↪seed 10 --i-metadata RDP_Tutorial_Metdata_file.txt --o-community-list community_ED3_
↪with_taxonomy.txt --i-distance-database RDP_distance_dictionary_20191126-150855.txt
```

3.3 Subsample Module

The subsample module is designed to take the final output community from the create module and provide a subsample of the community. The subsample module requires the input of the community generated from the create module. From here, the community can be subsampled to either include a specific number of strains (`--p-num-taxa`) or to represent specific proportions of a grouping variable (`--p-proportion`).

3.3.1 Subsample by total number of members to include

To subsample by number of members to include, we need to provide the output community from the create module (`--i-input`) and the number of strains to include in the final community (`--p-num-taxa`). We will limit our community to 100 members and also include a seed number for reproducibility.

```
>>> disco subsample --i-input-community community_ED3_with_taxonomy.txt --p-num-taxa␣
↪100 --p-seed 10
```

The above command should generate a tab delimited file that contains a list with only 100 community members that have a minimum of 3 nucleotide differences.

3.3.2 Subsample by proportions

To subsample by proportions of a grouping variable, we need to provide the output community from the create module (`--i-input`) and a file containing proportions of each group you wish to include (`--p-proportion`). We will subsample our community to reflect taxonomic proportions at the class level, of a natural microbiome and also include a seed number for reproducibility. We also need to indicate the column of the input community that we want to group by (here we use class).

```
>>> disco subsample --i-input-community community_ED3_with_taxonomy.txt --p-
↪proportion RDP_Tutorial_proportions_file.txt --p-seed 10 --p-group-by "Class"
```

3.4 18S example

Below is an example of how to create a community using an 18S aligned dataset and to subsample using the environment they were isolated from

```
>>> cd 18S_example/
>>> ls
```

```
18S_metadata.txt
18S_proportion_file.txt
18S_region_aligned.fasta
```

```
>>> disco create --i-alignment 18S_region_aligned.fasta --p-editdistance 3 --p-seed␣
↪10 --i-metadata 18S_metadata.txt --o-community-list community_ED3_18S.txt
>>> disco subsample --i-input-community community_ED3_18S.txt --p-proportion 18S_
↪proportion_file.txt --p-seed 10 --p-group-by Environment
```

3.5 Tutorial Completed

Congratulations! You have created a list of microbial strains that differ by at least 3 nucleotides. You then subsampled that list to either contain a specified number of strains or to reflect a specified proportion of groups. Please use the help option to view all options for the create and subsample modules.

```
>>> disco create -h
>>> disco subsample -h
```

CHAPTER

FOUR

FILE FORMAT DESCRIPTIONS

4.1 Create module

Input files

–i-alignment: This is an alignment of all sequences the user would like to evaluate in FASTA format

Example >S003715306

GCGGTA-AT-ACGTAG-GGAGCAAGCGTTGTC-CGG-ATTTATTGG-GCGTAAA-GAGCTCGTAG-G-CGGCTT-GGCAAGT-CGGATGTGAAA-CC-CCCAGG-CTTAACC-TGGGG-C–C- GCCATTCGA-TAC-TGC-TATGG-C-TT-GAGTTCGGTA-GGGGAT-TG-TGGA-ATT-CC-C-GGTGTAGCGGTGAAATGCGCAG-ATATCG-GG-AGGA-ACACC-AATG-GCGAAGGCAG- CAAT-CTGGGC-CGACACT-GA-CGCTGA-GG–A-GCGAAA-GCGTGGG-G-AGCAAA-CAGGATTAGATA

>S003614093

GCGGTA-AT-ACGTAG-GGAGCAAGCGTTGTC-CGG-AATTATTGG-GCGTAAA-GAGCTCGTAG-G-CGGTTC-GGTAAGT-CGGGTGTGAAA-AC-TCAAGG-CTCAACC-TTGAG-A–C- GCCACTCGA-TAC-TGC-CGTGA-C-TT-GAGTCCGGTA-GAGGAG-TG-TGGA-ATT-CC-T-GGTGTAGCGGTGAAATGCGCAG-ATATCA-GG-AGGA-ACACC-AGCG-GCGAAGGCGG- CACT-CTGGGC-CGGTACT-GA-CGCTGA-GG–A-GCGAAA-GCATGGG-G-AGCAAA-CAGGATTAGATA

>S001611178

GCGGTA-AT-ACGTAG-GGCGCGAGCGTTGTC-CGG-AATTATTGG-GCGTAAA-GGGCTCGTAG-G-CGGCTT-GTTGCGC-CTGCTGTGAAA-AC-GCGGGG-CTTAACT-CCGCG-C-GT- G-CAGTGGG-TAC-GGG-CA-GG-C-TT-GAGTGTGGTA-GGGGTG-AC-TGGA-ATT-CC-A-GGTGTAGCGGTGGAATGCGCAG-ATATCT-GG-AGGA-ACACCGAT-G-GCGAAGGCAG- GTCA-CTGGGC-CATTACT-GA-CGCTGA-GG–A-GCGAAA-GCGTGGG-T-AGCGAA-CAGGATTAGATA

–p-include-strains: A list of community members the community the user would like added. Each sequence identifier (must match what is in alignment)is on its own line

Example S003715306

S003614093

S001611178

–i-metadata: Information to combine with the community output. File must contain a header, be tab-delimited, and contain the identifiers in the first column

Example ID Phylum Class

S003715306 Actinobacteria Actinobacteria

S003614093 Actinobacteria Actinobacteria

S001611178 Actinobacteria Actinobacteria

S000014419 Actinobacteria Actinobacteria

–i-distance-database: Pre-calculated distance database of sequences, this is created when a previous create command has been run with the same strains. It is a tab delimited file with each line comparing two sequences. The sequence identifiers are in the first two columns and the edit distance between them is in the third

Example S003715306 S003614093 28

S003715306 S001611178 50

S003715306 S000014419 48

S003715306 S000015295 49

S003715306 S000022350 42

S003715306 S000129061 44

Output files

–o-community-list: A tab delimited list of strains, with each strain on its own line with a header line. If metadata is supplied it will be combined with this output

Example ID Phylum Class

S003715306 Actinobacteria Actinobacteria

S003614093 Actinobacteria Actinobacteria

S001611178 Actinobacteria Actinobacteria

S000014419 Actinobacteria Actinobacteria

–o-fasta: A FASTA file containing only the strains in the constructed community

4.2 Subsample module

Input files

–i-input-community: Tab seperated file with taxa ids in the first column with metadata in additional columns, output of create module

Example ID Phylum Class

S003715306 Actinobacteria Actinobacteria

S003614093 Actinobacteria Actinobacteria

S001611178 Actinobacteria Actinobacteria

S000014419 Actinobacteria Actinobacteria

–p-proportion: File of the relative proportions of each taxonomic rank desired in final community. Each rank is contained on its own line. The rank and the proportion are seperated by a tab.

Example Actinobacteria 0.1

Aquificae 0.001

Bacteroidia 0.05

Flavobacteriia 0.001

Sphingobacteriia 0.003

Output files

A file with each sequence identifier on its own lines for the subsampled community

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search