Supporting Text S3

PhyloPythiaS+: A Self-Training Method for the Rapid Reconstruction of Low-Ranking Taxonomic Bins from Metagenomes

Tutorial

This document describes how to use the *PhyloPythiaS+* software installed in a virtual machine.

Introduction – *PhyloPythiaS*

PhyloPythiaS (*PPS*) is a composition-based taxonomic binning method for metagenome samples that is based on structured support vector machines (SVMs). It works in three steps:

1. In the pre-processing phase, a human expert analyses an input metagenome sample, e.g. using the 16S analysis. S/he then decides which clades to model and what sample-derived data to use in the next step. This decision restricts the output space, i.e. a sequence from the metagenome sample will be assign to one of the modeled clades or to a corresponding clade at a higher taxonomic rank. Sample-derived data are sequences, usually directly from the metagenome sample (e.g. sequences that were found using the 16S analysis), with known labels. To model a clade, we need at least 100 kb of the sample-derived data or enough data in the reference sequence database (e.g. to model a particular clade at the genus rank, we usually require to have at least three genomes or draft genomes from three different species that belong to a particular clade.) Note that it is not recommended to model more than 100 clades, even though to model ~800 clades is possible.
2. In the training phase, training data are generated from the sample-derived and reference data for each modeled clade and subsequently used to train structured SVMs. The outputs of this step are trained models for the input clades. Note that this phase is usually time consuming and takes several hours depending on the number of modeled clades and training data used.
3. In the prediction phase, trained models are employed to assign the input metagenome sample. It is advantageous that once the models are build, they can be reused to predict several metagenome samples from the same environment. Note that sequences shorter than 1kb are not recommended to be assigned via *PhyloPythiaS* since they may not carry enough taxonomic information. This phase is substantially faster than the previous phase and takes typically only several minutes, depending mainly on the number of the input sequences.

Introduction – *PhyloPythiaS+*

PhyloPythiaS+ (*PPS+*) is an extension of *PhyloPythiaS* (*PPS*), which automates the first phase of

PPS, that used to be done manually, and adds an additional fourth evaluation phase. *PPS+* uses the marker gene analysis to determine which clades to model and what sample-derived data to use. It uses *HMMER* to search for marker genes and the naïve bayes classifier implemented in the *mothur* software as subprograms. The following marker genes are used: *16S*, *23S*, *5S*, *dnaG*, *infC*, *pgk*, *rpoB*, *tsf*, *frr*, *nusA*, *pyrG*, *rpmA*, *smpB*, *rpsC*, *rpsI*, *rpsK*, *rpsS*, *rpsB*, *rpsE*, *rpsJ*, *rpsM*, *rplA*, *rplB*, *rplC*, *rplD*, *rplE*, *rplF*, *rplK*, *rplL*, *rplM*, *rplN*, *rplP*, *rplS*, and *rplT*.

In the fourth evaluation phase, the following measures are computed:

- Scaffold-contig consistency. This measure tells how consistently contigs within corresponding scaffolds are assigned. (For real datasets if scaffold-contig mapping is available.)
- Confusion tables: taxonomic assignments of contigs vs assignments of the corresponding scaffolds. If the trained models are good, all the data should lie on the diagonal, i.e. contigs are assigned to the same clades as the corresponding scaffolds. (If sequences of contigs and scaffolds are available and if the scaffold-contig mapping is defined.)
- Training accuracy: assigned training data. If the trained models are good, the training data are assigned with high precision and recall. This is mainly used to evaluate the quality of the sample-derived data.
- Precision and recall measures are used to evaluate performance on simulated datasets.
- Confusion tables: assignments of contigs vs. reference assignments of contigs. Confusion tables are used to compare two assignments, where in the case both assignments are in agreement, all the data lie on the diagonal. (If reference assignments are available, e.g. for simulated datasets.)

Scenario

In this tutorial, you can analyze your own matagenome sample or you can analyze the provided test simulated dataset. This tutorial is divided into two parts. In the first part, you learn how to configure and run the *PPS+* pipeline. In the second part, after the *PPS+* pipeline is finished (which will take approx. 2-3hours), you learn how to analyze the results. You can use test datasets provided online: https://github.com/algbioi/datasets

Input data preparation

To run the pipeline, you need a FASTA file containing assembled contigs. Alternatively, long reads (i.e. > 1 kb), e.g. the high-quality PacBio consensus reads can be used as well. Optionally, you can provide also a FASTA file containing corresponding scaffolds and a tab separated scaffold-contig mapping file (first column ~ scaffold names, second column ~ contig names). If you are working with a simulated dataset, you should provide true labels for the contigs as a tab separated file (first column ~ contig name, second column ~ corresponding NCBI taxon id).

Configuration of *PPS+*

1. Go to a directory where you want to store the results of *PPS+*, e.g.:

```
cd /apps/pps/tests
```

2. Create a pipeline working directory that will later contain output and temporary files of the pipeline. (Note that the path to this directory is not allowed to contain '+' or '-' characters due to the mothur software), e.g.:

```
mkdir test01
```

3. The sample configuration files named config_ppsp_vm_refNCBI20121122_example.cfg and config_ppsp_vm_refNCBI20140513_example.cfg can be found in:

```
ls /apps/pps/tools
```

4. The configuration files differ by the reference used (i.e. NCBI20121122 or NCBI20140513), to view the files, type:

```
less /apps/pps/tools/config_ppsp_vm_refNCBI20121122_example.cfg
less /apps/pps/tools/config_ppsp_vm_refNCBI20140513_example.cfg
```

Type 'q' to quit.

5. Copy (cp), or rename (mv) one of the configuration file (depending on the reference you want to use), e.g.:

```
cp /apps/pps/tools/config_ppsp_vm_refNCBI20121122_example.cfg config_test01.cfg
```

6. Edit the configuration file, e.g. using vim:

```
vim config_test01.cfg
```

This is a list of all important configuration parameters that you should set (or keep default):

- pipelineDir … path to the pipeline directory (e.g. *CURRENT_DIR/test01*), that you have already created. (The path is not allowed to contain "-" or "+" due to the mothur software.)

INPUT FILES

- inputFastaFile … path to the FASTA file containing contigs to be classified (the path is not allowed to contain "-" or "+" due to the mothur software)
- inputFastaScaffoldsFile … path to the FASTA file containing scaffolds (optional)
- scaffoldsToContigsMapFile … path to the tab separated scaffold-contig mapping file (optional)
- referencePlacementFileOut … path to the tab separated reference placement file containing true labels for the simulated dataset or a reference binning (e.g. produced by a different binning method); the results of *PPS+* will be compared to this reference binning. (optional)

REFERENCE

If you use a simulated dataset, it is necessary to mask the corresponding reference sequences form

the reference database. This is necessary; otherwise you use the same sequence data in the training and predicting phase, which typically yields unrealistically good results. Usually, we exclude the reference data at the strain, species, or genus taxonomic ranks, which simulates new strains, species, or genera in the metagenome sample. E.g., if we exclude the reference data at the species rank, it means that we do not use any reference sequence from the species contained in the metagenome sample. To use this option, it is necessary that you specify the referencePlacementFileOut parameter; then, you just specify at which taxonomic rank you want to exclude the reference sequences.

- excludeRefSeqRank … at which taxonomic rank you want to exclude the reference sequences from the reference sequence database used in the training phase (note that the parameter referencePlacementFileOut must be defined if this option is used). Set it to e.g. *species*.
- excludeRefMgRank … at which taxonomic rank you want to exclude the reference sequences from the marker gene reference sequence database used in the marker gene analysis (note that the parameter referencePlacementFileOut must be defined if this option is used). Set it to e.g. *species*.

You should keep the default paths in this section unchanged.

TOOLS

You should keep the default paths in this section unchanged.

BASIC SETTINGS

- rankIdCut … you can specify down to which taxonomic rank the sample will be classified, where taxonomic ranks family(4), genus(5), or species(6) are recommended.
- maxLeafClades … the maximum number of leaf clades that will be considered in the classification. Only the most abundant clades, according to the marker gene analysis (or the most abundant clades in terms of the amount of reference sequences in the reference sequence databases in case of the general model) will be considered. You can set this number to the expected number of genera/species in the metagenome sample.
- minPercentInLeaf … if a leaf clade is assigned less data than this percentage by the marker gene analysis considering all data assigned to the leaf clades, such a leaf clade will not be considered.
- minSeqLen … sequences shorter than this length will not be classified since short sequences usually do not carry enough taxonomic information to be classified using composition based methods.

ADVANCED SETTINGS

- minGenomesWgs … if parameter rankIdCut from the BASIC SETTINGS is set to 6 (species), it is necessary to change this parameter (minGenomesWgs) to 1.

It is not recommended to change any other parameter in the advanced settings. You can find the description of the other parameters in the configuration file.

7. Make sure that you save all changes you have done to the configuration file.

Run *PPS+*

Now, after you have configured the pipeline, you can run it using one of the following commands.

To get all available options of the *PPS+* pipeline, run:

```
ppsp -h
```

To run only the marker gene analysis, enter the following command (Do not run this command if you want to do the whole analysis!).

```
ppsp -c config_test01.cfg -n -g -o s16 mg -s
```

To run the whole pipeline, enter the following command (assuming configuration parameter inputFastaScaffoldsFile was not set, i.e. you do not have the corresponding FASTA file containing scaffolds):

```
ppsp -c config_test01.cfg -n -g -o s16 mg -t -p c -r -s
```

To run the whole pipeline, enter the following command (assuming you set the configuration parameter inputFastaScaffoldsFile and scaffoldsToContigsMapFile, i.e. you have the corresponding FASTA file containing scaffolds and the scaffold-contig mapping file). Note that the command line parameter *"*-a*"* can be optionally used to compute the training accuracy.

```
ppsp -c config_ppsp_test01.cfg -n -g -o s16 mg -t -p c s v -r -s
```

Analysis of *PPS+* Results

After the pipeline is finished. You can examine the following output files; all important output files are contained in directory pipelineDir/output.

1. Assignment files

- inputFastaFile.fna.pOUT … a tab separated file containing assigned sequences, first column ~ contig name, second column ~ NCBI taxon id
- inputFastaFile.fna.PP.pOUT … assignment file in the *PhyloPythia* (*PP*) format (contains scientific names of the corresponding clades).

2. Abundance profiles

- profiles … A directory containing a CSV file for each taxonomic rank that contains respective abundance profile.

3. An overview of clades that were used to build the *PPS* models.

- summary_train.txt … A list of clades generated by the marker gene analysis, all clades contained in this list were modeled. 1St column ~ sum of sample specific data in terms of

base pairs (bp), 2nd column ~ sum of sample-derived data in terms of the number of sequences, 3rd column ~ scientific name (NCBI taxon id) of a clade at rank *superkingdom*, 4th column ~ corresponding *phylum* rank, 5th column corresponding *class* rank, etc. (Note that this file is a result of the marker gene analysis and does not reflect any change that you may have done manually to the list of clades or the sample specific data)

- summary_all.txt … a list of all clades for which some sample-derived data was found (note that only the most abundant clades from this list were modeled).

4. Precision and recall measures (are computed if the configuration parameter referencePlacementFileOut is specified in the configuration file, i.e. true labels are known, usually for simulated datasets). The precision/recall measures are computed either in terms of the number of sequences (count) or in terms of the number of base pairs (bp). Moreover, the precision/recall are computed considering different bin sizes (weighted) or not considering different bin sizes (not weighted), i.e. all bins are treated equally.

- precision_recall.csv … precision and recall measures computed at different taxonomic ranks considering referencePlacementFileOut as true labels of the input sequences (see also configuration parameters recallMinFracClade and precisionMinFracPred)
- precision_recall_no_ssd.csv … precision and recall computed only based on the input sequences different from the sequences used as sample-derived data.
- (precision_recall_corrections.csv … (see configuration parameter correctLabelThreshold))

5. Confusion tables – Comparison to reference assignments (if configuration parameter referencePlacementFileOut was set).

- cmp_ref … contains comparison tables for different taxonomic ranks, where rows correspond to the true assignments (referencePlacementFileOut from the configuration file) and columns correspond to the predicted contigs. Ideally, all the data lie on the diagonal.

6. Scaffold-contig consistency.

- inputFastaFile.fna.cons … the scaffold-contig consistency file (the consistency is computed based on the scaffoldsToContigsMapFile from the configuration file)

7. Confusion tables: Taxonomic assignments of contigs vs assignments of scaffolds. (Computed if configuration parameters inputFastaScaffoldsFile and scaffoldsToContigsMapFile were set.)

- contigs_vs_scaff … taxonomic assignments of contigs correspond to columns and assignments of scaffolds correspond to rows.

8. Training accuracy.

- train_accuracy … contains precision and recall (as well as comparison tables) for different training data types where the training data are considered as true assignments.
- train_accuracy/train_accuracy_all.txt … training accuracy of all the training data.
- train_accuracy/train_accuracy_sampleSpecificDir.txt … training accuracy of all training data generated from the sample-derived data. This corresponds to the quality of the sample-

derived data found by the marker gene analysis.

9. Log files

- log … contains log files for different subroutines (e.g. *PPS* train, *PPS* predict), if an error occurs, check the corresponding log file.

10. List of other files from the pipelineDir/working directory

- projectDir … *PPS* working directory.
- projectDir/models … after *PPS* training phase finishes, the models are stored in this directory and can be reused.
- ncbids.txt … the list of clades (taxa) generated by the marker gene analysis and used to train *PPS* models.
- sampleSpecificDir … the directory containing the sample specific data generated by the marker gene analysis and used to train *PPS* models.
- PPS_config_generated.txt … generated *PPS* configuration file.
- *.sl … large files containing temporary data (feature vectors) that can be removed after the whole pipeline finishes.