Supplementary Materials to "PhylOligo: a package to identify contaminant or untargeted organism sequences in genome assemblies."

Ludovic Mallet[1], Tristan Bitard-Feildel[2], Franck Cerutti[1], and Hélène Chiapello[1]

[1]INRA UR875, Unité Mathématiques et Informatique Appliquées de Toulouse (MIAT), Auzeville, 31326 Castanet-Tolosan, France
[2]CNRS UMR7590, Sorbonne Universités, Université Pierre et Marie Curie – Paris 6 – MNHN – IRD – IUC, Paris, France.

May 11, 2017

Contents

1 Workflow

PhylOligo is a package of tools to analyse the heterogeneity of oligonucleotide composition of genomic assembly fragments to explore and locate sequences from potential untargeted organisms. The package contains several programs arranged in a workflow.

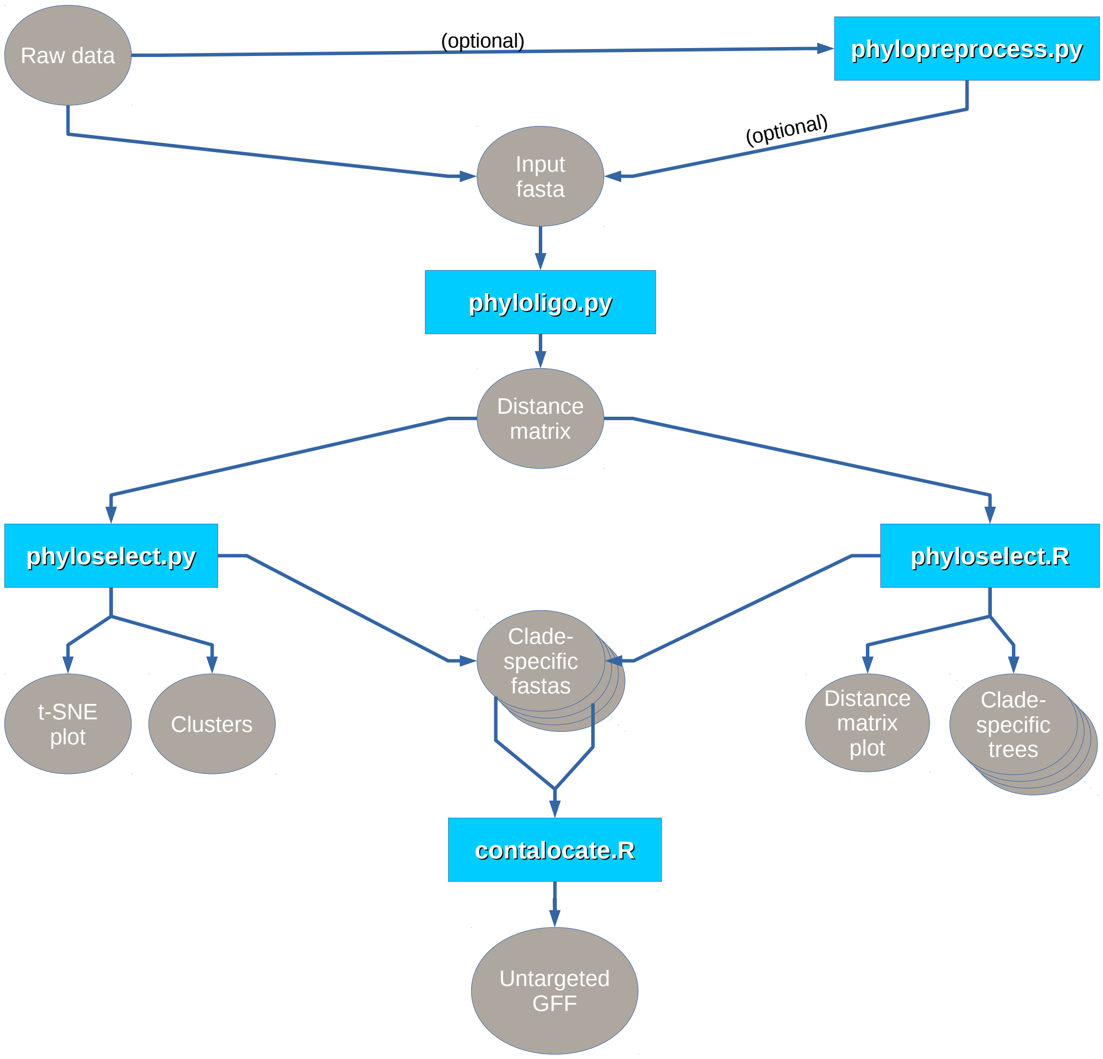

Figure 1: Workflow of PhylOligo. Blue frames: programs and scripts. Grey blobs: data files and output files.

2 Installation

PhyloOligo software needs python 3.4 or newer and several R and python packages.

2.1 Quick Install

Basic dependencies

If python or R are not installed on your system, call your distribution's package manager:

```
sudo apt-get install python3-dev python3-setuptools r-base git emboss samtools
#or
yum install python3-dev python3-setuptools r-base git emboss samtools
```

Clone/download the git repository

```
git clone https://github.com/itsmeludo/PhylOligo.git
```

or download it from https://github.com/itsmeludo/PhylOligo

Install python scripts and dependencies

If you have administrator rights or if you are working in a python virtual environment:

```
git clone https://github.com/itsmeludo/PhylOligo.git
cd PhylOligo
pip3 install .
```

You can also install it locally using:

```
git clone https://github.com/itsmeludo/PhylOligo.git
cd PhylOligo
pip3 install . --user
```

Or to install it locally in a folder of your choice:

```
pip3 install . --prefix /my/local/folder
```

If locally installed, be sure to add the local directory with executable in your executable path. On linux:

```
export PATH=$HOME/.local/bin:$PATH

phyloligo.py -h
```

2.2 Alternative install tricks

If the easy install procedure fails on your system, there are several options to install the dependencies.

Python requirements

If you want to install the dependencies separately use:

```
cd PhylOligo
pip3 install -r requirements.txt
```

Install R scripts and dependencies

In R, as root or user

```
R
install.packages(c("ape","getopt","gplots"))
```

Rights and paths

Link the programs into a directory listed in your $PATH

```
#cd PhylOligo

export PATH=`pwd`/src/:$PATH
chmod +x src/{*.py,*.R}
```

List of Dependencies:

- Python 3.x
  - BioPython biopython.org
  - sklearn http://scikit-learn.org/stable/install.html
  - Numpy numpy.org
  - matplotlib http://matplotlib.org
  - hdbscan https://pypi.python.org/pypi/hdbscan
  - Cython http://cython.org

  - h5py http://www.h5py.org
- R 3.x
  - ape http://ape-package.ird.fr
  - gplots https://cran.r-project.org/web/packages/gplots/index.html
  - getopt https://cran.r-project.org/web/packages/getopt/getopt.pdf
- EMBOSS http://emboss.sourceforge.net/download
- Samtools http://www.htslib.org/
- X11 onlyrequiredtorunphyloselect.R

3 Software manual and options

3.1 phylopreprocess.py

Pre-process the original contigs/scaffolds/long reads in order to filter out entries, reduce computational time and increase signal. Filter short sequences or highly conserved repeats. Sub-sampling can be used in order to perform quick tests or to reduce the size of a dataset to allow for its computation given the computational resources available. Note that this step is optional and that `phyloselect.R` also contains sequence filters in order to test out different values without having to recompute the frequencies and the distance matrix with `phyloligo.py`. Sequences shorter than 1kb should be considered as poorly informative or representative of their species compositional profile. In order to grant a more refined selection of materials to establish an accurate compositional profile prototype and the detection of potential untargeted sequences, sequences below about 5Kb could be filtered if it can be hypothesised that a possible contaminant would not have shorter sequences or be completely filtered out.

- Reads an assembly or long sequencing reads multi-fasta file
- Output filtered dataset

```
phylopreprocess.py [-h] -i INPUTFASTA [-p PERCENTILE] [-m MIN_SEQSIZE] [-c
    CUMULATED_SEQSIZE] [-s SAMPLING][-u SAMPLE_SIZE] [-r] [-o OUTPUTFASTA]
```

Parameters:

```
-h, --help               show this help message and exit
-i INPUTFASTA
-p PERCENTILE            remove sequences of size not in Xth percentile
-m MIN_SEQSIZE           remove sequences shorter than the provided minimal
                         size
-c CUMULATED_SEQSIZE     select sequences until their cumulated size reach this
                         parameter. if -r is used, sequences are picked
                         randomly.
-s SAMPLING              percentage of reads to sample
-u SAMPLE_SIZE           number of reads to sample
-r                       the order of the sequences are randomized
-o OUTPUTFASTA
```

3.2 phyloligo.py

Generate the all-by-all contig distance matrix

- Load and index the genome assembly sequences.
- Compute the kmer/spaced-pattern composition profile of each sequence in the assembly.
- Compute a pairwise distance matrix for all sequences.

```
phyloligo.py -d JSD -i genome.fasta -o genome.JSD.mat -u 64
```

Parameters:

```
  -h, --help            show this help message and exit
  -i GENOME, --assembly GENOME
                        multifasta of the genome assembly
  -k PATTERN, --lgMot PATTERN
                        word lenght / kmer length / k [default:4]. This option
                        is an alias for --pattern (see -p). If the type of the
                        parameter is an integer, it will be interpreted as the
                        lenght of the kmer to use. If the type of the
                        parameter is a string, it will be interpreted as a
                        spaced-pattern.
  -s {both,plus,minus}, --strand {both,plus,minus}
                        strand used to compute microcomposition.
                        [default:both]
  -d {Eucl,JSD}, --distance {Eucl,JSD}
                        how to compute distance between two signatures : Eucl
                        : Euclidean[default:Eucl], JSD : Jensen-Shannon
                        divergence
  --freq-chunk-size FREQCHUNKSIZE
                        the size of the chunk to use in scoop to compute
                        frequencies
  --dist-chunk-size DISTCHUNKSIZE
                        the size of the chunk to use in scoop to compute
                        distances
  --method {scoop,joblib}
                        don't use scoop to compute distances use joblib
  --large {None,memmap,h5py}
                        used in combination with joblib for large dataset
  -c THREADS_MAX, --cpu THREADS_MAX
                        how many threads to use for windows microcomposition
                        computation[default:4]
  -o OUT_FILE, --out OUT_FILE
                        output file[default:phyloligo.out]
  -w WORKDIR, --workdir WORKDIR
                        working directory
  -p PATTERN, --pattern PATTERN
                        spaced-word pattern string, only containing 1s and 0s,
                        i.e. '100101001', default='1111'. See -k / --lgMot.
```

3.3 phyloselect.R

Regroup contigs by compositional similarity on a tree and explore the topology.

- Load the distance matrix produced by PhylOligo.
- Optionally create a hierarchically sorted distance matrix.
- Build a cladogram from the distance matrix.
- Interactively ask the user to explore the cladogram and select clads that might correspond to untargeted sequences based on the interpretation of the topology.
- Export clad-specific fasta files:
  - To inspect their potential origin for example with blast or GOHTAM (Ménigaud *et al.*, 2012)
  - To use as learning material in ContaLocate

```
phyloselect.R -d -m -c 0.95 -s 4000 -t BIONJ -f c -w 20  -i genome.JSD.mat -a
    genome.fasta -o genome_conta
```

Parameters:

```
    -i|--matrix
                    All-by-all contig distance matrix, tab separated (required)
    -a|--assembly
                    Multifasta file of the contigs (required)
    -f|--tree_draw_method
                    Tree building type. [phylogram, cladogram,
                    fan, unrooted, radial] by default cladogram.
    -t|--tree_building_method
                    Tree drawing type [NJ, UPGMA, BIONJ, wardD,
                    wardD2, Hsingle, Hcomplete, WPGMA, WPGMC, UPGMC] by default NJ.
    -m|--matrix_heatmap
                    Should a matrix heatmap should be produced
    -c|--distance_clip_percentile
                    Threshold to exclude very distant contigs based on the distance
                    distribution. Use if the tree is squashed by repeats or
                    degenerated/uninformative contigs [0.97]
    -s|--contig_min_size
                    Min length in bp of contigs to use in the matrix and tree.
                    Use if the tree is squashed by repeats or
                    degenerated/uninformative contigs [4000]
    -d|--dump_R_session
                    Should the R environment be saved for later exploration?
                    The filename will be generated from the outfile parameter
                    or its default value
    -g|--max_perc
                    Max edge assembly length percentage displayed (%)
    -l|--min_perc
                    Min edge assembly length percentage displayed (%)
    -k|--keep_perc
                    Ratio of out-of-range percentages to display (%)
    -o|--outfile
                    Outfile name, default:phyloligo.out
    -b|--branchlength
                    Display branch length
    -w|--branchwidth
                    Branch width factor [40]
    -v|--verbose
                    Says what the program do.
    -h|--help
                    This help.
```

note: PhyloSelect uses the library Ape and its interactive clade selection function on a tree plot with the mouse. X11 is therefore required. If the program has to run on a server -typically for memory reasons- please use the -X option of ssh to allow X11 forwarding.

3.4 phyloselect.py

Regroup contigs by compositional similarity: hierarchical DBSCAN or K-medoids clustering and multidimensional scaling display with t-SNE.

- Load the distance matrix produced by PhylOligo.
- Cluster the sequences
- Export cluster-specific fasta files:
  - To inspect their potential origin for example with blast or GOHTAM (Ménigaud *et al.*, 2012)
  - To use as learning material in ContaLocate

```
phyloselect.py -i genome.JSD.mat -t -m hdbscan --noX -o genome_conta
```

Parameters:

```
-h, --help              show this help message and exit
-i DISTMAT              The input matrix file
-t                      Perform tsne for visualization and pre-clustering
-p PERPLEXITY           Change the perplexity value
-m {hdbscan,kmedoids}
                        Method to use to compute cluster on transformed
                        distance matrix
--minclustersize MIN_CLUSTER_SIZE
                        Set the minimal cluster size of an HDBSCAN cluster
--minsamples MIN_SAMPLES
                        Set the minimal sample size of an HDBSCAN cluster
-k NBK                  Number of cluster
-f FASTAFILE            Path of the original fasta file used for the
                        computation of the distance matrix
--interactive           Allow the user to run the script in an interactive
                        mode and change clustering parameter on the fly
                        (require -t)
--large {memmap,h5py}
                        Used in combination with joblib for large dataset
--noX                   Instead of showing pictures, store them in png
-o OUTPUTDIR
```

3.5 contalocate.R

Extract DNA segments with homogeneous oligonucleotide composition from a genome assembly. Once you have explored your assembly's oligonucleotide composition, identified and selected -potentially partial- untargeted genome material, use ContaLocate to target species-specific DNA according to a double parametrical threshold.

- Learn a compositional profile for the host and the untargeted organism, previously identified with phyloligo.py / phyloselect.R.
- Scan the assembly for regions similar in composition to the two aforementioned profiles.
- Compute one threshold value for each scan based on the distribution of the metric.
- Locate the untargeted regions according to the 2 thresholds, distant from the host and close the the untargeted profile.
- Generate a GFF3 map of the untargeted region positions in the genome.

If both the host and untargeted learning material are avaialble:

```
contalocate.R -i genome.fasta -r genome_host.fa -c genome_conta_1.fa
```

The training set for the host genome can be omitted if the amount of untargeted sequences is negligible/very small. In this case, the profile of the host will be trained on the whole genome, including the untargeted sequences which might create a bias proportional to the relative amount of untargeted material.

```
contalocate.R -i genome.fasta -c genome_conta_1.fa
```

The set up of the thresholds can be manually enforced. The user will interactively prompted to set the thresholds given the distribution of windows divergence.

```
contalocate.R -i genome.fasta -c genome_conta_1.fa -m
```

Parameters:

```
-i|--genome
                         Multifasta of the genome assembly (required)
-r|--host_learn
                        Host training set (optional)
-c|--conta_learn
                        Contaminant training set (optional) if missing and
                        sliding window parameters are given, the sliding
                        windows composition will be compared to the whole
```

```
                         genome composition to contrast potential HGTs
                         (prokaryotes and simple eukaryotes only)
    -t|--win_step
                         Step of the sliding windows analysis to locate the
                         contaminant (optional) default: 500bp or 100bp
    -w|--win_size
                         Length of the sliding window to locate the
                         contaminant (optional) default: 5000bp
    -W|--outputdir
                         path to outputdir directory
    -d|--dist
                         Divergence metric used to compare profiles: (KL), JSD or Eucl
    -m|--manual_threshold
                         You will be asked to manually set the thresholds
    -h|--help
                         This help
```

4 Pipeline examples

4.1 Workstation

```
assembly=/path/to/assembly.fa
cpus=64
name="organism"
pattern=4
distance="JSD"
work_dir=`pwd`



phyloligo.py -c $cpus -o ${name}_${distance}_k${pattern}.mat -i $assembly -k
    $pattern -d ${distance} --method joblib

phyloselect.R -i ${name}_${distance}_k${pattern}.mat -a $assembly -d -w 20 -c
    0.90 -s 4000 -m -f c -o PhyloSelect_${name}



# filenames depends on the selection made by the user.
contalocate.R -i genome.fasta -r PhyloSelect_${name}_1.fa -c PhyloSelect_${name}
    _2.fa
```

4.2 SGE grid - SMP

```
#!/bin/bash

assembly=/path/to/assembly.fa
cpus=64
name="organism"
pattern="1111"
distance="JSD"
work_dir=`pwd`


#$ -S /bin/bash
#$ -cwd
#$ -V
#$ -pe parallel_smp $cpu
#$ -l mem=1G
#$ -l h_vmem=1G
#$ -N PhylOligo_grid_test_$name



echo "phyloligo.py -c \$NSLOTS -o ${name}_${distance}_k${pattern}.mat -i
    $assembly --pattern $pattern -d ${distance} --method joblib --large h5py" |
    qsub -N PhylOligo_${name}_${distance}_k${pattern} -l mem=12G -l h_vmem=64G
```

```
echo "phyloselect.py -i ${name}_${distance}_k${pattern}.mat -t -m hdbscan --
    large h5py --noX -o $work_dir"| qsub -N PhyloSelect_${name} -l mem=10G -l
    h_vmem=30G -hold_jid PhylOligo_${name}_${distance}_k${pattern}
```

4.3 SGE grid - Multi node

```
#!/bin/bash

assembly=/path/to/assembly.fa
cpus=64
name="organism"
pattern="1111"
distance="JSD"
work_dir=‘pwd‘


#$ -S /bin/bash
#$ -cwd
#$ -V
#$ -pe parallel_smp $cpu
#$ -l mem=1G
#$ -l h_vmem=1G
#$ -N PhylOligo_grid_test_$name

#SSH connexion between nodes must be allowed for scoop to work properly

echo "phyloligo.py -c \$NSLOTS -o ${name}_${distance}_k${pattern}.mat -i
    $assembly --pattern $pattern -d ${distance} --method scoop --freq-chunk-size
     3000 --dist-chunk-size 500" | qsub -N PhylOligo_${name}_${distance}_k${
    pattern} -l mem=12G -l h_vmem=64G


echo "phyloselect.py -i ${name}_${distance}_k${pattern}.mat -t -m hdbscan --noX
    -o $work_dir"| qsub -N PhyloSelect_${name} -l mem=10G -l h_vmem=30G -
    hold_jid PhylOligo_${name}_${distance}_k${pattern}
```

4.4 SGE grid - Very large dataset

```
#!/bin/bash

assembly=/path/to/assembly.fa
cpus=240
name="organism"
pattern="1111"
distance="JSD"
work_dir=‘pwd‘


#$ -S /bin/bash
#$ -cwd
#$ -V
#$ -pe parallel_smp $cpu
#$ -l mem=1G
#$ -l h_vmem=1G
#$ -N PhylOligo_grid_test_$name

echo "phylopreprocess.py -i $assembly -m 4000 -o ${assembly}_filtered_m4000.fa"
    | qsub -N PhylOligo_${name}_${distance}_k${pattern}_preprocess -l mem=12G -l
     h_vmem=64G

echo "phyloligo.py -c \$NSLOTS -o ${name}_${distance}_k${pattern}.mat -i ${
    assembly}_filtered_m4000.fa --pattern $pattern -d ${distance} --method
    joblib --large h5py" | qsub -N PhylOligo_${name}_${distance}_k${pattern} -l
    mem=48G -l h_vmem=100G -hold_jid PhylOligo_${name}_${distance}_k${pattern}
    _preprocess


echo "phyloselect.py -i ${name}_${distance}_k${pattern}.mat -t -m hdbscan --
    large h5py --noX -o $work_dir"| qsub -N PhyloSelect_${name} -l mem=800G -l
    h_vmem=3000G -hold_jid PhylOligo_${name}_${distance}_k${pattern}
```

5 Examples

5.1 *Magnaporthe oryzae*

This example shows how bacterial regions were identified in assemblies of the phytopathogenic fungus *Magnaporthe oryzae*. Nine isolates were sequenced (Chiapello *et al.*, 2015) of which four exhibited an unexpectedly larger genome size compared to other genomes of the same species. This dataset can be directly downloaded at http://genome.jouy.inra.fr/gemo/data_publi/genome/TH12-rn_prefix.fna

Investigations with PhylOligo and comparison of the different isolates (Figure 2) revealed the presence of a subset of contigs with distinct oligonucleotide composition as seen on Figure 3 (Clade B) and Figure 4. These regions were used to learn a prototype of this composition, determine divergence thresholds (see Figure 5) and the whole genome was scanned with ContaLocate. A whole bacterial genome was identified in 3 out of the 9 isolates, as well as several chimeric scaffolds, *i.e.* containing DNA from 2 organisms. Using Blast (Altschul *et al.*, 1997) and GOHTAM (Ménigaud *et al.*, 2012), the bacterial genome was identified to be unsequenced at the time and compositionally close to *Burkholderia phytofirmans* and *Burkholderia xenovorans*.

The genome of *Magnaporthe oryzae* after filtering was not significantly altered as suggested by analyses performed with BUSCO and DOGMA presented in Table 1 and Table 2 respectively.

	Raw assembly		Filtered assembly	
Complete BUSCOs (C)	1249	95.0%	1248	94.9%
Complete and single-copy BUSCOs (S)	1243	94.5%	1244	94.6%
Complete and duplicated BUSCOs (D)	6	0.5%	4	0.3%
Fragmented BUSCOs (F)	61	4.6%	62	4.7%
Missing BUSCOs (M)	5	0.4%	5	0.4%
Total BUSCO groups searched	1315		1315	

Table 1: Genome annotation statistics performed with BUSCO (Simão *et al.*, 2015) before and after filtering with PhylOligo on *Magnaporthe oryzae*. Reference set used: ascomycota_odb9.

Size of Conserved Domain Arrangements	Raw assembly	Filtered assembly	Filtered Bacteria
1 domain	97.26%	95.73%	98.11%
2 domains	93.87%	91.91%	94.77%
3 domains	90.91%	88.81%	90.24%
Total	95.77% (1470/1535)	94.07% (1444/1535)	96.63% (545/564)

Table 2: Statistics of the expected protein domain arrangements in the genome annotation performed with DOGMA (Dohmen *et al.*, 2016) before and after filtering with PhylOligo on *Magnaporthe oryzae* and the extra materials from *Burkholderia*. Percentages denote the completeness of expected domain arrangements present. The default expected set was used for *Magnaporthe oryzae*. The Bacterial expected set was used for *Burkholderia*.

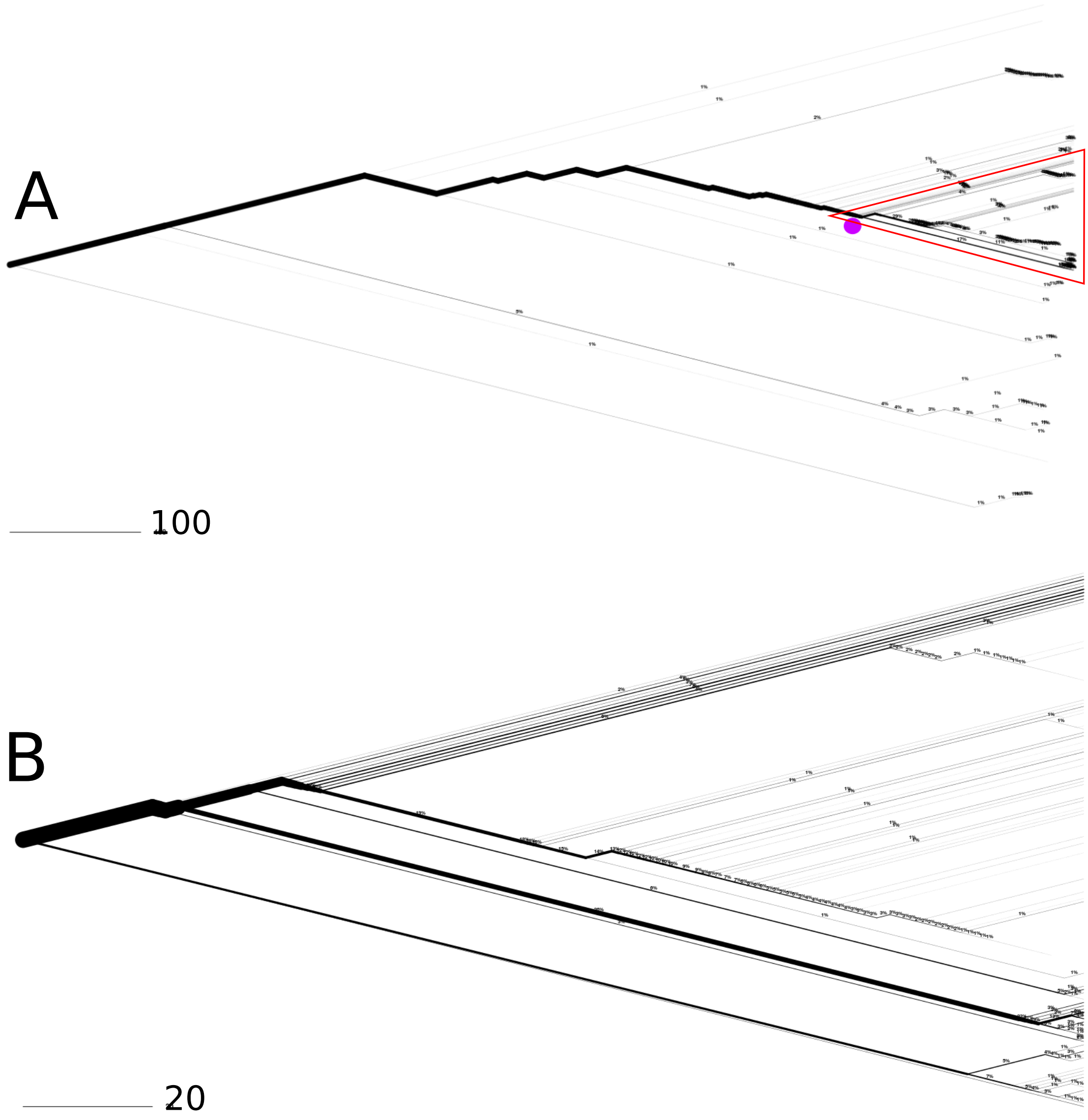

Figure 2: Exploration of 2 isolates of *Magnaporthe oryzae* (Chiapello *et al.*, 2015). **A**: Exploration of *Magnaporthe oryzae* TH12. The topology and width pattern of the subtree identified in the red triangle is very similar to the whole tree in Figure 2 B. The user suspects that this conserved pattern accounts for the targeted organism, and that the extra clades might represent untargeted sequences, as the clade banches very early on the cladogram and represent a small amount of sequences in the assembly. **B**: Exploration of *Magnaporthe oryzae* TH16. This isolate was found to contain no untargeted material.

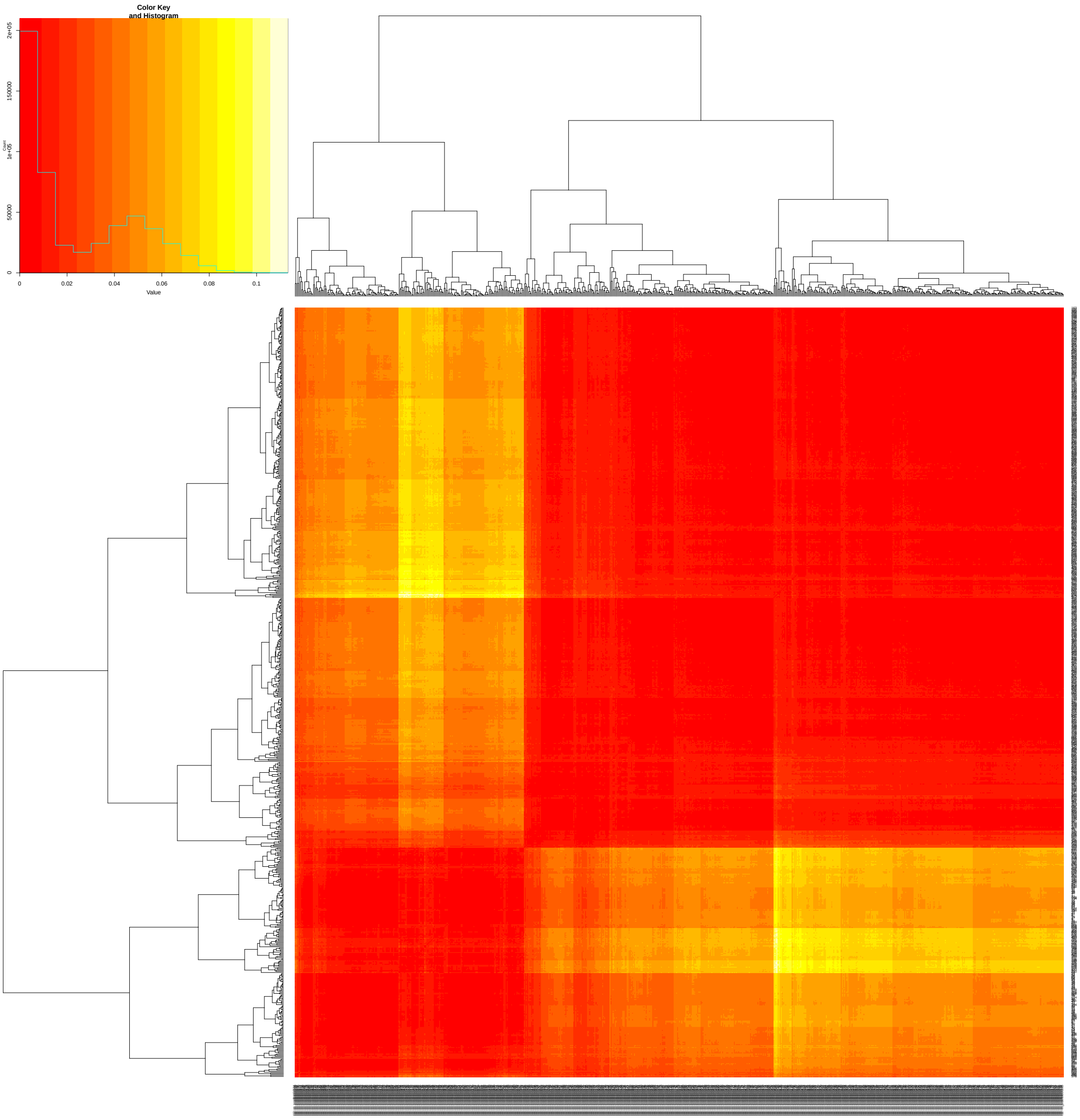

Figure 3: Sorted distance matrix of contigs of *Magnaporthe oryzae* TH12 assembly (Chiapello *et al.*, 2015). The following parameters in phyloselect.R were used: -c 0.97 -m

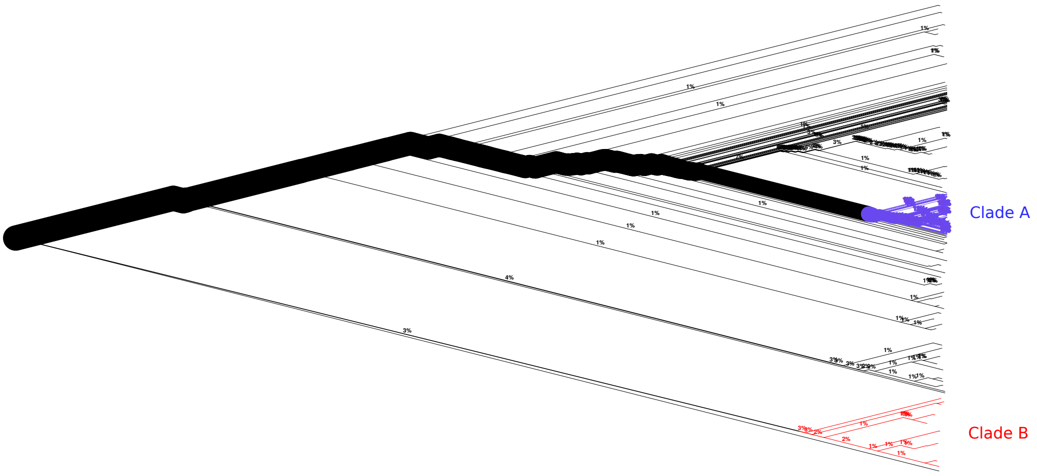

Figure 4: Exploration of the *Magnaporthe oryzae* TH12 assembly (Chiapello *et al.*, 2015). The width of the branches is set proportional to the cumulative size of contigs in the sub tree. The thicker path on the tree indicates a set of contigs with homgeneous oligonucleotide composition cumulating the majority of the assembled sequences The selection in blue will be called "Clade A", the user suspect this correspond to the host sequences, *Magnaporthe oryzae*. The selection in red will be called "Clade B", the user suspect this is an untargeted set of sequences, as the clade banches very early on the cladogram and represent a small amount of sequences in the assembly.

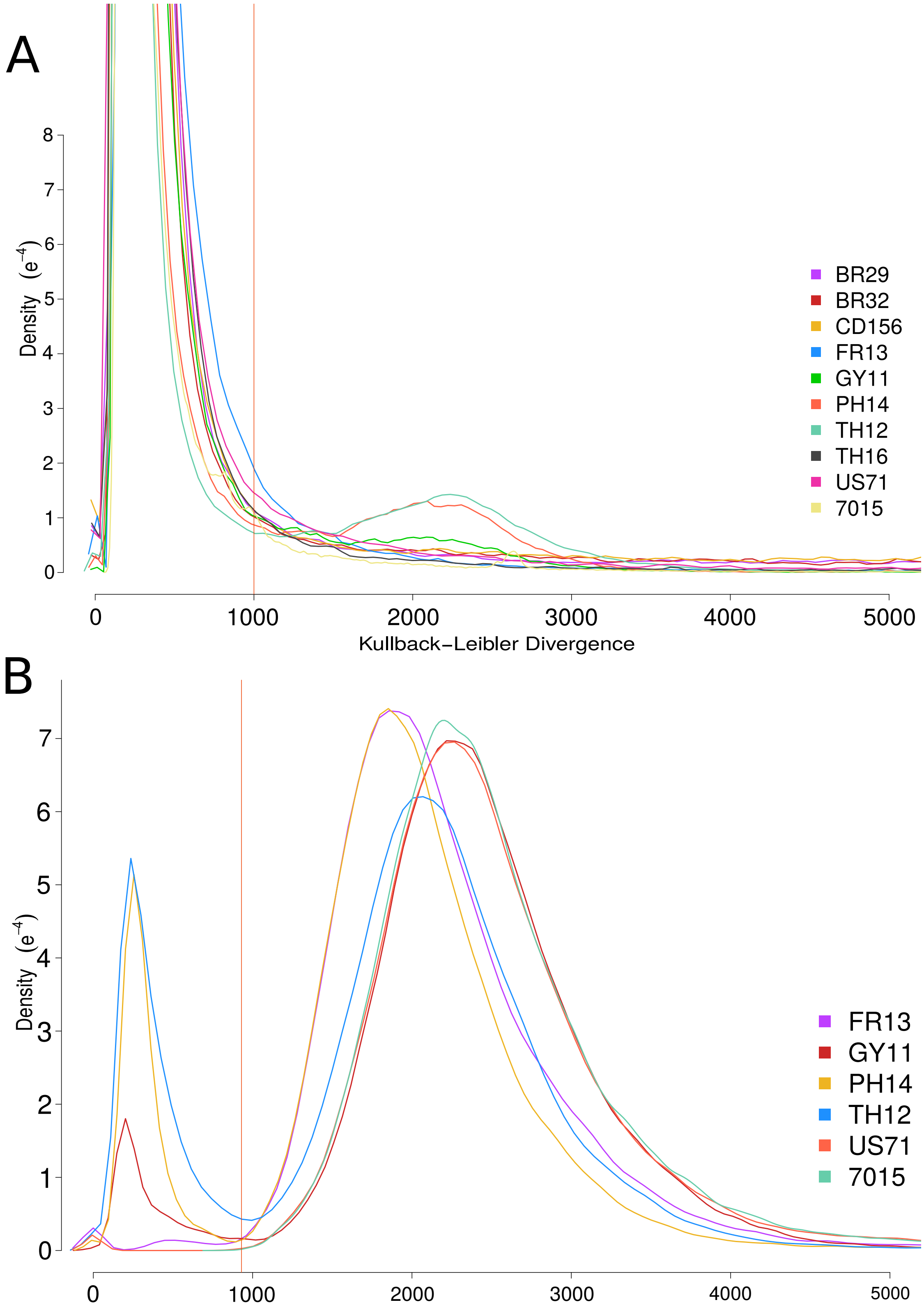

Figure 5: **A**: Distribution of distances between the composition profile of clade A in Figure 4 and the scanning windows over the whole assembly. The host threshold is the vertical red line. Each coloured curve is a *Magnaporthe oryzae* isolate from (Chiapello *et al.*, 2015). **B**: Distribution of distances between the composition profile of clade B in Figure 4 and the scanning windows over the whole assembly. The untargeted threshold is the vertical red line.

5.2 *Hypsibius dujardini*

The recent sequencing of the tardigrade (Boothby *et al.*, 2015) yielded a controversy about the composition of its genome sequence. Running PhylOligo on the genome assembly revealed the presence of sets of contigs with an homogeneous oligonucleotide composition grouping in diverging clades (Figure 6 A, clades *not* in red), which is in agreement with the previously proposed multiple contamination of the sample (Delmont and Eren, 2016; Koutsovoulos *et al.*, 2016). Unlike the example of *Magnaporthe oryzae*, the cladogram displays many branching clades each containing a substantial fraction of the assembled data. these dataset can be retrived from `https://www.ncbi.nlm.nih.gov/Traces/wgs/?val=LMYF01&display=contigs&page=1` and `https://www.ncbi.nlm.nih.gov/Traces/wgs/?val=LRSR01#contigs`.

We comparatively ran PhylOligo on the assembly proposed by (Delmont and Eren, 2016) (Figure 6 B) which was filtered based on several criteria including the presence of known bacterial genes and kmer composition. The kmer composition based tree obtained with the filtered assembly can be identified as a rather conserved subtree in the original assembly composition tree (red triangle in Figure 6 A). This observation supports the ability of PhylOligo to display atypically branching groups of contigs on a compositional basis as an evidence for the presence of untargeted sequences.

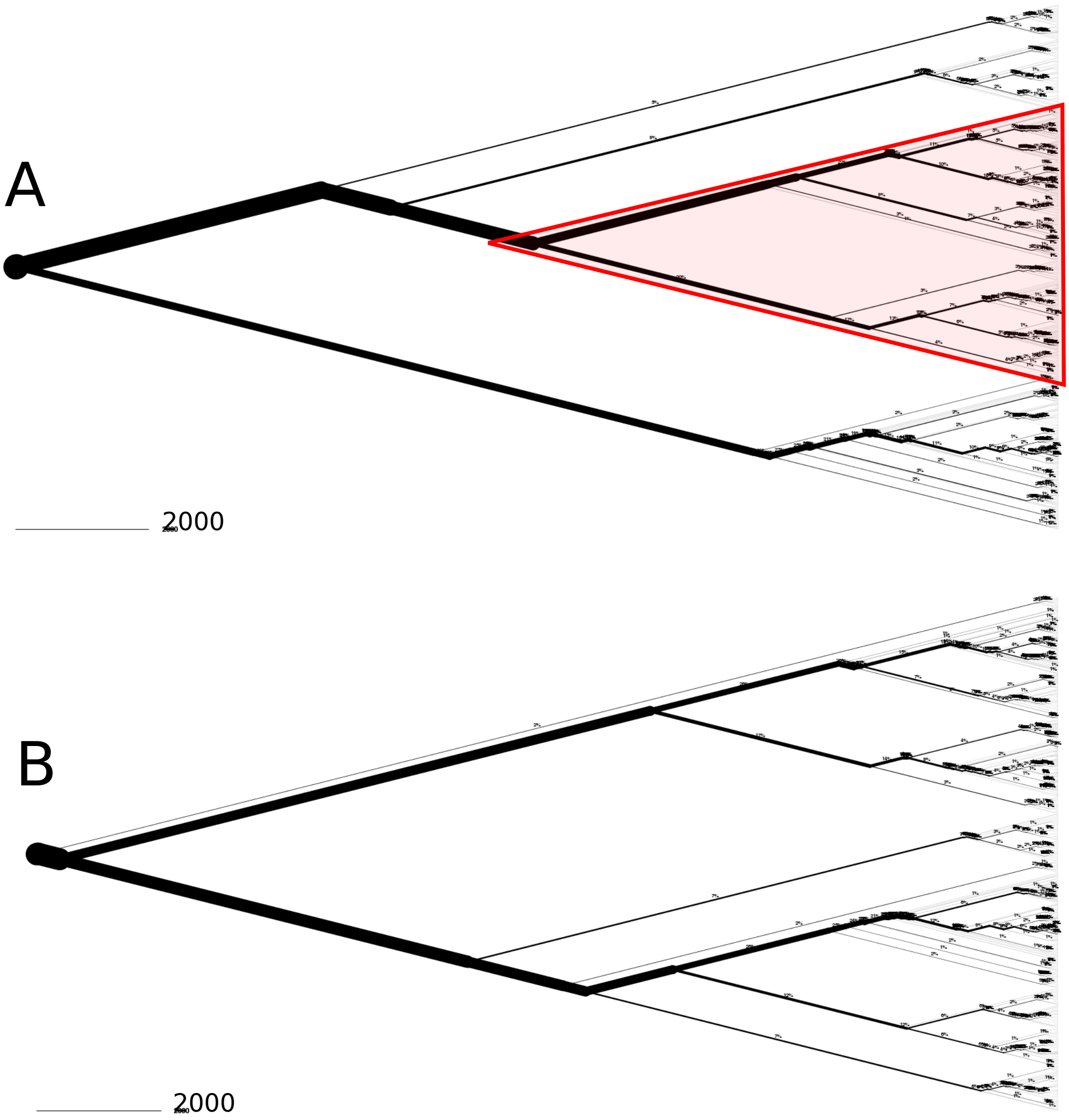

Figure 6: **A**: Exploration of the *Hypsibius dujardini* original assembly (Boothby *et al.*, 2015). **B**: Exploration of the *Hypsibius dujardini* curated assembly (Delmont and Eren, 2016). A very similar tree topology and branch width pattern can be identified in the original assembly (Figure 6 A) in the red triangle.

6 Benchmark and simulations

6.1 Methods

To assess the performance of PhylOligo on varying conditions and situations, we generated artificial contaminations and quantified the specificity and sensitivity of contaminant sequences identification and clustering to generate the learning material geared to set the thresholds.

6.1.1 Genomes and assembly simulation

32 genomes (Table 3) were manually selected and downloaded from RefSeq (Ref). From the assembled sequenced, a draft quality was simulated by using GRINDER (Angly *et al.*, 2012) using short sequence simulation with a contig length following a normal distribution with parameters μ=10Kb and σ=10Kb. Contigs were sampled until the breadth of coverage of the initial genome reached 0.98.

Binomial name	Short name	Domain	URL
Archaeoglobus fulgidus	Aful	archaea	Archaeoglobus_fulgidus
Desulfurococcus fermentans	Dfer	archaea	Desulfurococcus_fermentans
Ferroglobus placidus	Fpla	archaea	Ferroglobus_placidus
Halococcus thailandensis	Htha	archaea	Halococcus_thailandensis
Haloferax mediterranei	Hmed	archaea	Haloferax_mediterranei
Pyrococcus furiosus	Pfur	archaea	Pyrococcus_furiosus
Thermococcus eurythermalis	Teur	archaea	Thermococcus_eurythermalis
Escherichia coli	Ecol	bacteria	Escherichia_coli
Staphylococcus aureus	Saur	bacteria	Staphylococcus_aureus
Burkholderia mallei	Bmal	bacteria	Burkholderia_mallei
Xanthomonas oryzae	Xory	bacteria	Xanthomonas_oryzae
Salmonella enterica	Sent	bacteria	Salmonella_enterica
Bacillus cereus	Bcer	bacteria	Bacillus_cereus
Aspergillus fumigatus	Afum	fungi	Aspergillus_fumigatus
Encephalitozoon cuniculi	Ecun	fungi	Encephalitozoon_cuniculi
Podospora anserina	Pans	fungi	Podospora_anserina
Saccharomyces cerevisiae	Scer	fungi	Saccharomyces_cerevisiae
Schizosaccharomyces pombe	Spom	fungi	Schizosaccharomyces_pombe
Yarrowia lipolytica	Ylip	fungi	Yarrowia_lipolytica
Giardia intestinalis	Glam	protozoa	Giardia_intestinalis
Leishmania major	Lmaj	protozoa	Leishmania_major
Paramecium tetraurelia	Ptet	protozoa	Paramecium_tetraurelia
Plasmodium falciparum	Pfal	protozoa	Plasmodium_falciparum
Trichomonas vaginalis	Tvag	protozoa	Trichomonas_vaginalis
Danio rerio	Drer	vertebrate other	Danio_rerio
Xenopus tropicalis	Xtro	vertebrate other	Xenopus_tropicalis
Apteryx australis	Aaus	vertebrate other	Apteryx_australis
Marmota marmota	Mmar	vertebrate mammalian	Marmota_marmota
Panthera tigris	Ptig	vertebrate mammalian	Panthera_tigris
Castor canadensis	Ccan	vertebrate mammalian	Castor_canadensis
Felis catus	Fcat	vertebrate mammalian	Felis_catus
Homo sapiens	Hsap	vertebrate mammalian	Homo_sapiens

Table 3: Species used in the benchmark

.

6.1.2 Contamination simulation

An all-by-all contamination assay of the 32 genomes was undertaken mixing a declared contaminant and host genome. The contaminant contigs were randomly sampled from the initial simulated draft genome up until a count of 1000 or the all the contigs of the draft if lower. The Host genomes was generated in the same manner with the exception of the maximum number of contigs raised to 2000. For all species combination, a contaminated set was generated and tested.

6.1.3 Automated PhylOligo pipeline

PhylOligo was tuned to run in automatic mode using `phyloselect.py` with the unsupervised H-DBSCAN. We discarded contigs shorter than 4Kb and performed several runs for each contaminated set with varying parameters as described in the next subsection. After the clustering step, we compute for each cluster the specificity for contaminant, *i.e.* the fraction of contaminant sequences in this cluster, the sensitivity *i.e.* the fraction of the whole contaminant draft genome aggregated in the cluster, and a hybrid score which is the product of sensitivity and specificity. In the context of selecting qualitative material to perform the learning step, we target a high specificity with the broader associated sensitivity, hence, the sensitivity matrix reports the value of the cluster with the highest specificity (displayed in the specificity matrix). In other terms, the two matrixes, specificity and sensitivity, display for each cell the relevant value for the same cluster. From all the clusters, the one with the higher specificity is selected and reported, ties in specificity are resolved by picking among the ties the cluster highest sensitivity. The Hybrid score matrix presents the cluster with the best computed value (specificity · sensitivity) which represents a trade-off and the advantage of minimising the weight of highly-specific-yet-very-small clusters.

6.2 Results

6.2.1 Kmers & patterns

We evaluated the discriminative power of several lengths and patterns for kmers on manually selected simulated contaminations. Simulated data included a panel of distant and closely related contaminations. The results are presented on Figure 7.

High specificity values (first graph) are to be interpreted as a result of the H-DBSCAN clustering, favouring homogeneous clusters. For mixes containing very closely related species, (*e.g. Salmonella enterica* in *Escherichia coli*), specificity varies with the kmers, the best values being obtained with k=111001 and 11111. Values for 111111 are somewhat lower in several cases, which is linked to the length of simulated contigs, the length of k and the subsequent lower esperance E (see subsection 7.1)

The sensitivity of a cluster is informative regarding it's ability to constitute a good learning material to set up the thresholds. The better the sensitivity, the more intra-genomic variability taken into account. The sensitivity displayed on Figure 7 (labelled 'Fraction', second graph) is somewhat coherent for all tested kmers excepted for 111111 for the reasons mentioned above. The best average values over all the sample contaminations are achieved for kmers 11001 to 11111. Across all the sample contamination the most robust kmer -higher median- is the kmer 110101 which is able to work well both on close and distant contaminants. It is to be noted that the historic 1111 is a working compromise, especially when having to deal with short fragments (see subsection 7.1), but seems to be lagging for close contaminations. In this case, the kmer 11001 stood for a better alternative.

The hybrid score (Figure 7, third graph), being maxed by the lowest value among specificity · sensitivity, is informative of the quality of the training material. A value one 1 means that the whole contaminant was clustered into one single cluster, so that the learning step will be performed on the whole contaminant. High values are achieved for the more distant contaminants. The most robust kmers across all the simulated samples are 111001 (excepted for close contaminants) and 110101.

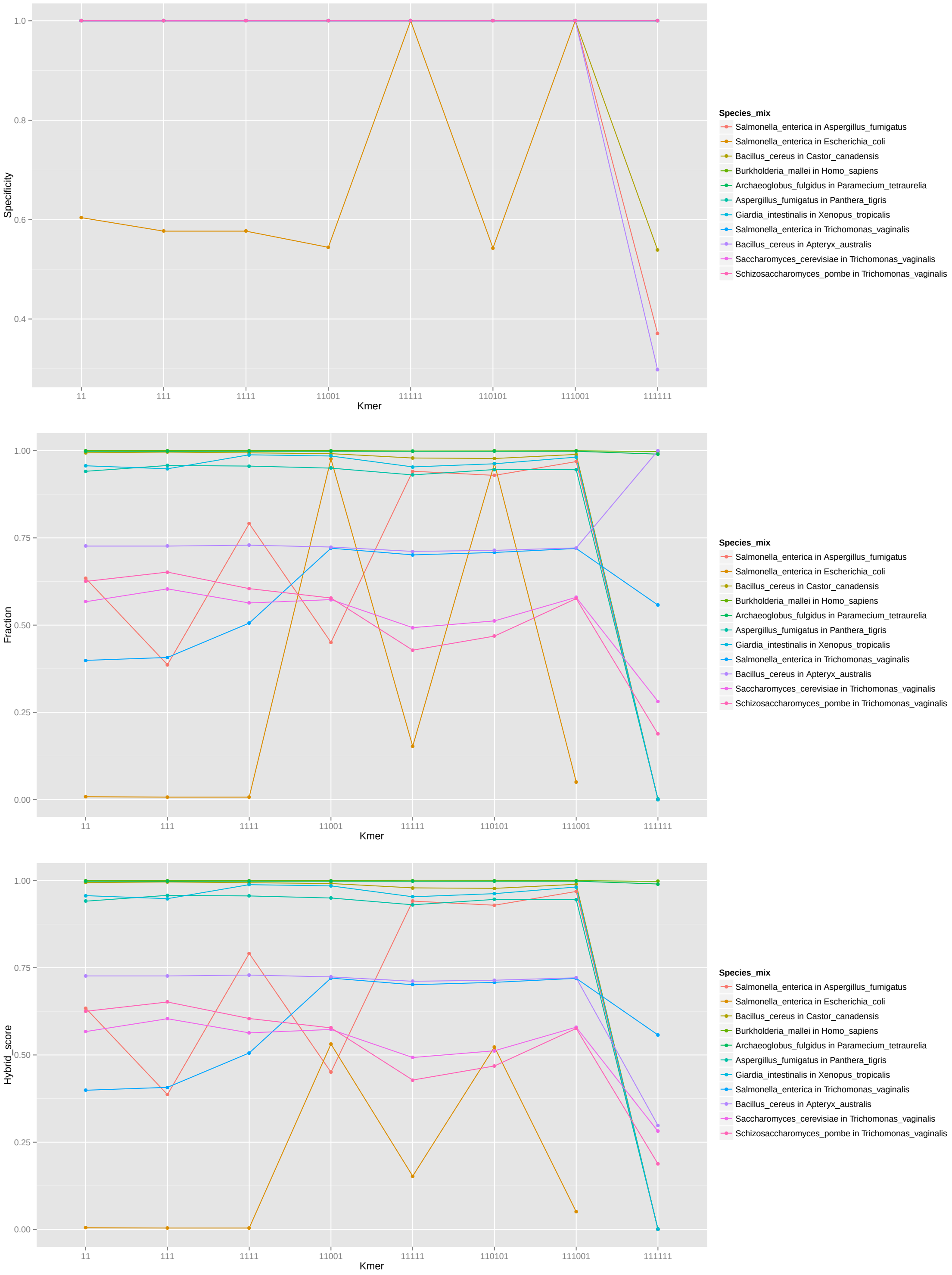

Figure 7: From top to bottom: Best specificity in a cluster, sensitivity in the cluster with highest specificity (Best sensitivity for ties on specificity), hybrid score.

6.2.2 Species spectrum

We selected 32 genomes across the domains of life with an emphasis on maximising diversity, spanning varying degrees of complexity, genome content, length and composition to assess the ability of PhylOligo and PhyloSelect to automatically identify clusters of contaminant sequences. We computed the statistics presented in subsubsection 6.1.3 for the all-by-all organism matrix contamination assay. Results are presented on Figure 8, Figure 9 and Figure 10.

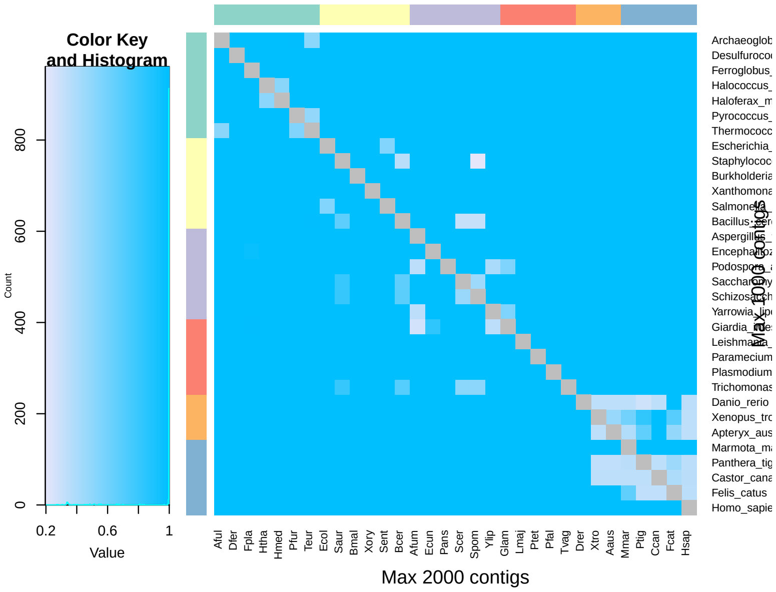

Figure 8: Best specificity in a cluster for pairwise contaminations.

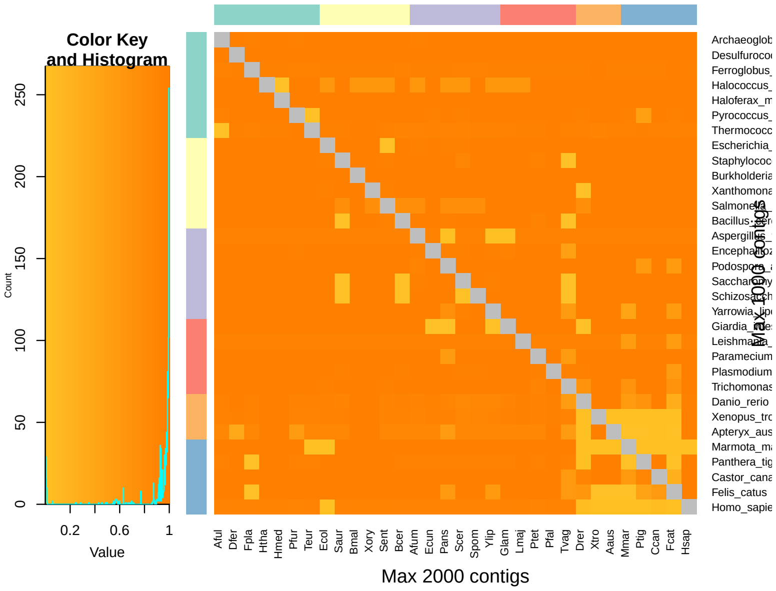

Figure 9: Sensitivity in the cluster with highest specificity (Best sensitivity for ties on specificity) for pairwise contaminations.

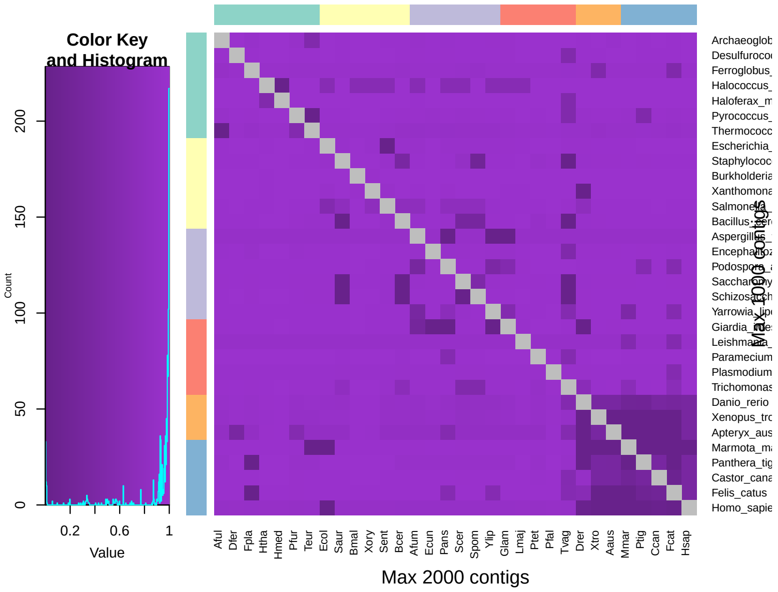

Figure 10: Hybrid score for pairwise contaminations.

6.3 Discussion

Overall, the benchmark demonstrate a great ability to discriminate contaminant clusters with very high specificity and good sensitivity, suited with the requirements for supervised learning and partitioning. Most contaminant contigs are clustered in almost perfect groups, highly sensitive and specific.

Some species mixes gave results where contaminant clusters would not contain the main part of the contaminant sequences. However, in all the cases a very specific cluster was found and gathered enough material to grant the learning step of `ContaLocate`.

To assess the boundaries of the method, we designed a conundrum case and selected a mix of extremely close species: *Escherichia coli* and *Salmonella enterica*. In this case we note that the choice of the kmer pattern plays a major role in separating the species sequences and constituting a contaminant-specific material large enough to train `ContaLocate`, but it was however the case for 2 kmers: 11111 and 111001.

The all-by-all mix of the 32 species simulated contigs revealed similar conclusions, with additional information regarding the 'vertebrate in vertebrate' sets, where it appeared more difficult to segregate species sequences with the automated pipleline and shorter kmers, which is consistent with the literature (Deschavanne *et al.*, 1999).

7 Discussion and strategies

7.1 Parameters

Several criteria should be taken into account in order to adapt the parameters of PhylOligo to match a datasets singularity.

Let's approximate the esperance E of occurrence of each oligonucleotide for a sequence of length L and a oligonucleotide of length k as:

$$E = (L - k + 1)/4^k$$

with $(L - k + 1)$ being the number of overlapping oligonucleotides of length k observed in a sequence of length L and 4^k being the number of different oligonucleotides of length k defined from $\{A, C, G, T\}$. See Reinert *et al.* (2000) for a better model.

Empirically, values of E above 10 (20 and up is a safer choice) are able to perform efficiently, to limit the noise from small counts and constrained discretisation of observed frequencies.

Because L and its distribution is a static data, it is recommended to adapt k accordingly. While longer oligonucleotides allow for a better species-specific profile, sufficiently fine granularity of the profile sampling must be achieved (sufficient E) therefore limited by L. With the current expectation of contig length in assemblies, the parameter k in PhylOligo should typically range between 3 and 5, 4 being a renowned trade-off and a common value in the literature (Ménigaud *et al.*, 2012; Kumar *et al.*, 2013; Alneberg *et al.*, 2014; Crusoe *et al.*, 2015; Eren *et al.*, 2015; Koutsovoulos *et al.*, 2016)

Some empirical analysis of the parameters k and L can be found in Deschavanne *et al.* (2000) and a thorough analysis of the probabilistic and statistical properties of words is developed in Reinert *et al.* (2000).

Overall true composition divergence between the targeted and untargeted organisms impacts the ability to cluster them apart, as the closer these organisms are in term of composition, the more difficult they are to distinguish. The metric used is also to be considered. More can be found in this benchmark paper: Becq *et al.* (2010) where they assessed the impact of various parameters and conditions including species proximity in the context of detecting horizontal transfers.

7.2 Limits and special cases

Organism proportions and assembly quality

Fragmentation and relative genome size and stoechiometry of dna at sequencing can impact how one should look for untargeted materials in their data. In Phyloselect.R, we emphasise on two features of the compositional structure within an assembly: the cumulative size of sequences in a clade, as defined by the width of a branch on the composition tree and the number of sequences grouping in a clade thus defining its breadth at the leaves. The respective proportion of contigs within a clade is displayed by the branch width on the cladogram, naturally making the larger untargeted more obvious. We designed this feature in order to cope with the possible differential fragmentation of the organisms: a higher fragmentation of the untargeted genome can make the clade to look more populated (broader at the leaf) but without impact on the branch width (cumulated contig size). This way, spotting an untargeted materials is more based on its relative size to the targeted genome and the distance between them than the fragmentation of their respective genome assembly. Accordingly, the missed untargeted materials are expected to be the smaller genomes, leading to an overall minimised error. Regardless of the proportion, in our experience the untargeted sequences tend to branch out on the most basal part of the cladogram, which is the observed case when the host genome and the untargeted organism are distant. In the case where a significant genomic part of only one organism would be missing (*i.e.* from filtering short contigs), identifying the matching clade can be less obvious. This should however have a limited impact on the overall filtering given enough learning material can be sampled (see section 7.2 "Training materials requirements and impact").

Training materials requirements and impact

The design of our tool and the learning step of PhyloOligo allow us to keep consistent results even when a small fraction of the untargeted data is interactively selected. This is possible because oligonucleotide signatures tends to be conserved along genomes (Dufraigne *et al.*, 2005), thus making untargeted organism sequences clustering in the same clade, even for lowly prevalent contaminant in a huge dataset. We recommend for this reason that the user trains ContaLocate using sequences of cumulated length of minimum 50kb and up (100 kb and up is a safer choice), the longer the better, even if the contaminant has a much larger whole genome. The first partition should be quite exhaustive, but thanks to the double threshold system, if the targeted and untargeted organisms are far apart enough, most of the untargeted should be detected. It is moreover possible to perform a second iteration of the partition using the result of the first (untargeted material) as learning material. The second run should take into account the most of the intra-genomic variability of the untargeted genome and improve the results.

Intragenomic compositional variability

Genomes composition exhibits different standards across the realms of life. Some organisms tend to have a genome homogeneous all along, while some can contain compositional segments or isochores (Deschavanne *et al.*, 1999; Bernardi *et al.*, 1985; Cuny *et al.*, 1981) with variable degrees of divergence. Intra- and inter- chromosomal compositional divergence is observed in several organisms, especially in complex eukaryotes. In this case, the conditions for a user to be misled by the representation of the compositional exploratory tree proposed in PhylOligo would require that the compositional profile of the untargeted materials fall within the variation observed across the chromosomes, which although possible, remains extraordinarily specific.

In the case of bacteria, compositionally divergent Islands are well known of which some are thought to be horizontally transferred (see the reference of Pierneef *et al.* (2015) for an a web-service to explore islands detected from tetranucleotide divergence). These Islands should be no problem at all if they are assembled with the rest of the targeted genome. Otherwise, they might appear not clustering well with the targeted contigs. Even in this case, one would nonetheless expect the island to appear with a thin branch because of its relative size to the genome, and not necessarily with a basal branching since the island's composition might have started to converge towards the composition of the host upon accumulation of mutations since its acquisition). Lastly, even the isolated islands would likely not be detected as contaminant because of the double threshold system, unless the user did specifically select this clade and assumed it might be a contaminant.

In general, training materials should be selected to best each represent the variability of either the targeted and untargeted genomes. (see section 7.2 "Training materials requirements and impact")

Chimeric sequences

Assemblies performed with sequencing material mixing reads from both targeted and untargeted organisms might contain chimeric scaffolds or even contigs. Chimeric sequences are expected to branch on the compositional tree in-between the targeted and untargeted clades, as a function of the proportion of material from each. These chimeric sequences should be split after a run of contalocate.R given that a representative amount of each composition profile was sampled for the training.

Chimeric sequences can become problematic when their proportion within the assembly and the spectrum of species-proportion within contigs would constitute a continuum on the tree making the visualisation and selection of species-specific clades for accurate composition learning difficult to the user. This possibility can be evaluated *a posteriori* by asserting that the 2 modes on the distribution of distances between genome windows and both the prototype profiles of the targeted and untargeted organisms are well separated (each distribution do not overlap significantly) and not too broad (see Figure 5). If these criteria are not met, the suggested procedure is to run (a few) iterations of PhylOligo + Contalocate in order to perform the learning on a ever-enriched and more species-specific material, by splitting the initial data at the positions suggested by ContaLocate (gff file). An alternative assembly strategy or further investigation on the quality of the sequencing libraries might also be considered.

Repeated content and transposable elements

Transposable elements (TEs) can bloom in genomes over evolutionary time causing a wide range of changes. In some species, embedded contraptions of TEs can be formed, easily reaching dozen of Kb in length and hard to assemble even with the longer PacBio reads, especially for TE families that underwent a recent expansion upon a so-called TE burst, usually assorted with low to medium diversity. These long repeated structures often accumulate repetitive sequences, often inactivated and/or partial copies from one or few TEs families, of which a large part appear to be species-specific and which can exhibit a oligonucleotide composition different from that of its host genome. TE content can be dominant in some organisms and become the most fragmented part, yet the most represented in contigs or scaffold in the assembly, depending on the evolutionary time of the burst, the presence of passive or active (RIP, *etc.*) TE ageing, the sequencing material and the performance of the assembler software. Such repeated contigs, by the multi factorial conditions and the impact on the assembly structure can easily imped the discovery of untargeted material for the following reasons:

- The proportion of TE-filled contigs is significant and disrupt the reading of the targeted sequences along the compositional tree, as the branch width splits into big groups, potentially not close from the the targeted genome clade (for some TE families). Especially if these contig tetranucleotide composition is different from the rest of the genome.
- Compositional clades can be over-representing TEs making relevant targeted and untargeted barely visible in interactive exploration.

- If an active TE inactivation system is in place, such as RIP, which can impact tetranucleotide composition.
- For tools using the depth of coverage (PhylOligo doesn't), if the assembler over-merged the numerous copies into a very few in the final assembly, the information of coverage is skewed and can lead to a false identification of untargeted materials.

The suggested approach to this is to comprehend the TE content, by identifying the known and species specific families in at least the targeted organism and see on which clade(s) of the tree these would match best using blast for example. Alternatively, we implemented a set of filters in `phyloselect.R` (-c float[0-1]) and `phylopreprocess.py` (-p PERCENTILE) allowing the user to exclude very close (low-distance) sequences based on the distribution of the distances in the distance matrix. A percentile of distances can be specified and the contigs exhibiting a mean distance to all other contigs within under this percentile will be excluded, effectively removing the most repeated sequences from the dataset. As mentioned in the manual of `phylopreprocess.py` (subsection 3.1), it should be considered best practice to remove contigs from the dataset by `phylopreprocess.py` best in last resort to achieve computation of the distance matrix and then experiment with various filter values in `phyloselect.R` to visualise their impact on the exploration.

Sliding window genome scan

The accuracy of the boundaries defined by Contalocate are limited by the step of the sliding window system, by default 500bp. This can be adjusted by using a shorter step, dramatically increasing computational time and rising the possibility to create very short positive regions in an on-then-off pattern around the true position that we describe as Dentelle- or lacework-like. The size of the window matters, as contigs shorter than the windows size will be considered homogeneous in composition by ContaLocate as the overlapping scan can't be informative. Such contigs will hence not be split if they were chimeric. The length of the sliding window should be chosen according to the oligonucleotide length as stated in subsection 7.1. As a rule of thumb, recommended window size for k=3,4,5,6 ranges, respectively, between 1000-1500bp, 4000-5000bp, 15000-20000bp and 50000-100000bp.

8 Bibliography

Alneberg, J., Bjarnason, B. S., de Bruijn, I., Schirmer, M., Quick, J., Ijaz, U. Z., Lahti, L., Loman, N. J., Andersson, A. F., and Quince, C. (2014). Binning metagenomic contigs by coverage and composition. *Nat Meth*, **11**(11), 1144–1146. Brief Communication.

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. (1997). Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Research*, **25**(17), 3389.

Angly, F. E., Willner, D., Rohwer, F., Hugenholtz, P., and Tyson, G. W. (2012). Grinder: a versatile amplicon and shotgun sequence simulator. *Nucleic Acids Research*, **40**(12), e94.

Becq, J., Churlaud, C., and Deschavanne, P. (2010). A benchmark of parametric methods for horizontal transfers detection. *PLOS ONE*, **5**(4), 1–9.

Bernardi, G., Olofsson, B., Filipski, J., Zerial, M., Salinas, J., Cuny, G., Meunier-Rotival, M., and Rodier, F. (1985). The mosaic genome of warm-blooded vertebrates. *Science*, **228**(4702), 953–958.

Boothby, T. C., Tenlen, J. R., Smith, F. W., Wang, J. R., Patanella, K. A., Osborne Nishimura, E., Tintori, S. C., Li, Q., Jones, C. D., Yandell, M., Messina, D. N., Glasscock, J., and Goldstein, B. (2015). Evidence for extensive horizontal gene transfer from the draft genome of a tardigrade. *Proceedings of the National Academy of Sciences*, **112**(52), 15976–15981.

Chiapello, H., Mallet, L., Guérin, C., Aguileta, G., Amselem, J., Kroj, T., Ortega-Abboud, E., Lebrun, M.-H., Henrissat, B., Gendrault, A., Rodolphe, F., Tharreau, D., and Fournier, E. (2015). Deciphering genome content and evolutionary relationships of isolates from the fungus Magnaporthe oryzae attacking different host plants. *Genome Biology and Evolution*, **7**(10), 2896–2912.

Crusoe, M. R., Alameldin, H., Awad, S., Boucher, E., Caldwell, A., Cartwright, R., Charbonneau, A., Constantinides, B., Edvenson, G., Fay, S., Fenton, J., Fenzl, T., Fish, J., Garcia-Gutierrez, L., Garland, P., Gluck, J., González, I., Guermond, S., Guo, J., Gupta, A., Herr, J., Howe, A., Hyer, A., Härpfer, A., Irber, L., Kidd, R., Lin, D., Lippi, J., Mansour, T., McA'Nulty, P., McDonald, E., Mizzi, J., Murray, K., Nahum, J., Nanlohy, K., Nederbragt, A., Ortiz-Zuazaga, H., Ory, J., Pell, J., Pepe-Ranney, C., Russ, Z., Schwarz, E., Scott, C., Seaman, J., Sievert, S., Simpson, J., Skennerton, C., Spencer, J., Srinivasan, R., Standage, D., Stapleton, J., Steinman, S., Stein, J., Taylor, B., Trimble, W., Wiencko, H., Wright, M., Wyss, B., Zhang, Q., zyme, e., and Brown, C. (2015). The khmer software package: enabling efficient nucleotide sequence analysis. *F1000Research*, **4**(900).

Cuny, G., Soriano, P., Macaya, G., and Bernardi, G. (1981). The major components of the mouse and human genomes. *European Journal of Biochemistry*, **115**(2), 227–233.

Delmont, T. O. and Eren, A. M. (2016). Identifying contamination with advanced visualization and analysis practices: metagenomic approaches for eukaryotic genome assemblies. *PeerJ*, **4**, e1839.

Deschavanne, P., Giron, A., Vilain, J., Dufraigne, C., and Fertil, B. (2000). Genomic signature is preserved in short dna fragments. In *Proceedings of the 1st IEEE International Symposium on Bioinformatics and Biomedical Engineering*, BIBE '00, pages 161–, Washington, DC, USA. IEEE Computer Society.

Deschavanne, P. J., Giron, A., Vilain, J., Fagot, G., and Fertil, B. (1999). Genomic signature: characterization and classification of species assessed by chaos game representation of sequences. *Molecular Biology and Evolution*, **16**(10), 1391.

Dohmen, E., Kremer, L. P., Bornberg-Bauer, E., and Kemena, C. (2016). Dogma: domain-based transcriptome and proteome quality assessment. *Bioinformatics*, **32**(17), 2577.

Dufraigne, C., Fertil, B., Lespinats, S., Giron, A., and Deschavanne, P. (2005). Detection and characterization of horizontal transfers in prokaryotes using genomic signature. *Nucleic Acids Research*, **33**(1), e6.

Eren, A. M., Esen, O. C., Quince, C., Vineis, J. H., Morrison, H. G., Sogin, M. L., and Delmont, T. O. (2015). Anvi'o: an advanced analysis and visualization platform for 'omics data. *PeerJ*, **3**, e1319.

Koutsovoulos, G., Kumar, S., Laetsch, D. R., Stevens, L., Daub, J., Conlon, C., Maroon, H., Thomas, F., Aboobaker, A. A., and Blaxter, M. (2016). No evidence for extensive horizontal gene transfer in the genome of the tardigrade hypsibius dujardini. *Proceedings of the National Academy of Sciences*, **113**(18), 5053–5058.

Kumar, S., Jones, M., Koutsovoulos, G., Clarke, M., and Blaxter, M. (2013). Blobology: exploring raw genome data for contaminants, symbionts and parasites using taxon-annotated gc-coverage plots. *Frontiers in Genetics*, **4**, 237.

Ménigaud, S., Mallet, L., Picord, G., Churlaud, C., Borrel, A., and Deschavanne, P. (2012). Gohtam: a website for 'genomic origin of horizontal transfers, alignment and metagenomics'. *Bioinformatics*, **28**(9), 1270–1271.

Pierneef, R., Cronje, L., Bezuidt, O., and Reva, O. N. (2015). Pre_gi: a global map of ontological links between horizontally transferred genomic islands in bacterial and archaeal genomes. *Database*, **2015**, bav058.

Reinert, G., Schbath, S., and Waterman, M. S. (2000). Probabilistic and statistical properties of words: An overview. *Journal of Computational Biology*, **7**, 1–46.

Simão, F. A., Waterhouse, R. M., Ioannidis, P., Kriventseva, E. V., and Zdobnov, E. M. (2015). Busco: assessing genome assembly and annotation completeness with single-copy orthologs. *Bioinformatics*, **31**(19), 3210.