dbcAmplicons pipeline Bioinformatics

Matthew L. Settles

Genome Center Bioinformatics Core

University of California, Davis

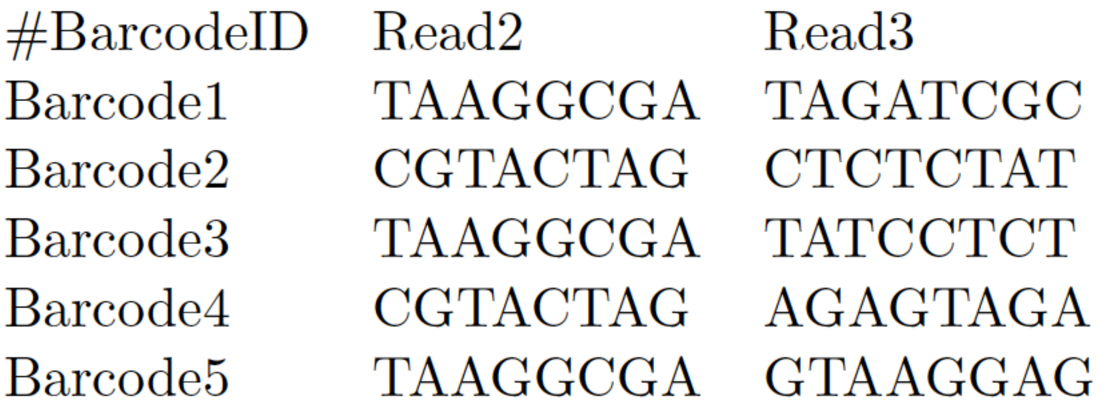
settles@ucdavis.edu; bioinformatics.core@ucdavis.edu

Workshop dataset: Slashpile

- Slash Pile – Accumulated debris from cutting brush or trimming trees
    - Measured, bacteria and fungal communities using 5 amplicons
        - 16sV1V3
        - 16sV4V5
        - ITS1
        - ITS2
        - LSU
- 3 – slashpiles
- 2 depths
- Distance from slashpile

Input Files: Barcode Table

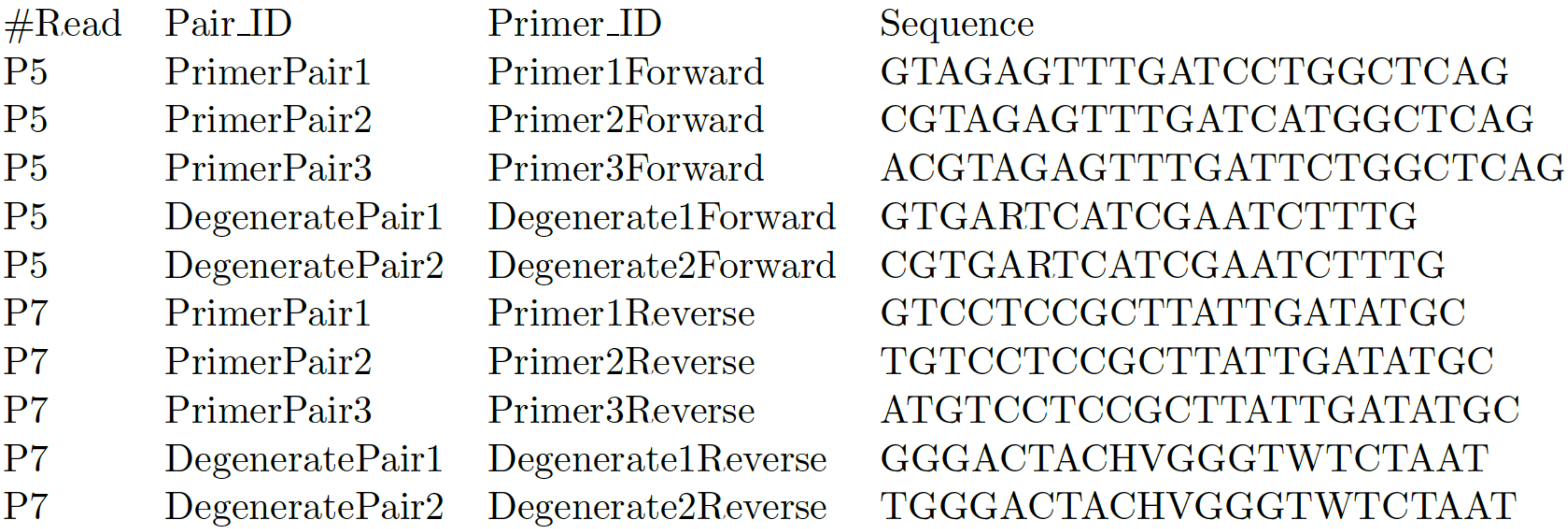
Requires 3 columns: BarcodeID [a name for the pair], Index1 (Read2 in RC), Index2 (Read3) in a plain tab-delimited text file. Orientation is important, but you can change in the preprocess arguments. First line is a comment and just help me remembers.



#BarcodeID	Read2	Read3
Barcode1	TAAGGCGA	TAGATCGC
Barcode2	CGTACTAG	CTCTCTAT
Barcode3	TAAGGCGA	TATCCTCT
Barcode4	CGTACTAG	AGAGTAGA
Barcode5	TAAGGCGA	GTAAGGAG

Input Files: Primer Table

Requires 4 columns: the read in which the primer should be checked for (allowable are P5/P7, R1/R2, READ1/READ2, F/R, FORWARD/REVERSE, Primer Pair ID describes which should be found 'together', Primer ID individual id, and sequence (IUPAC ambiguity characters are allowed).

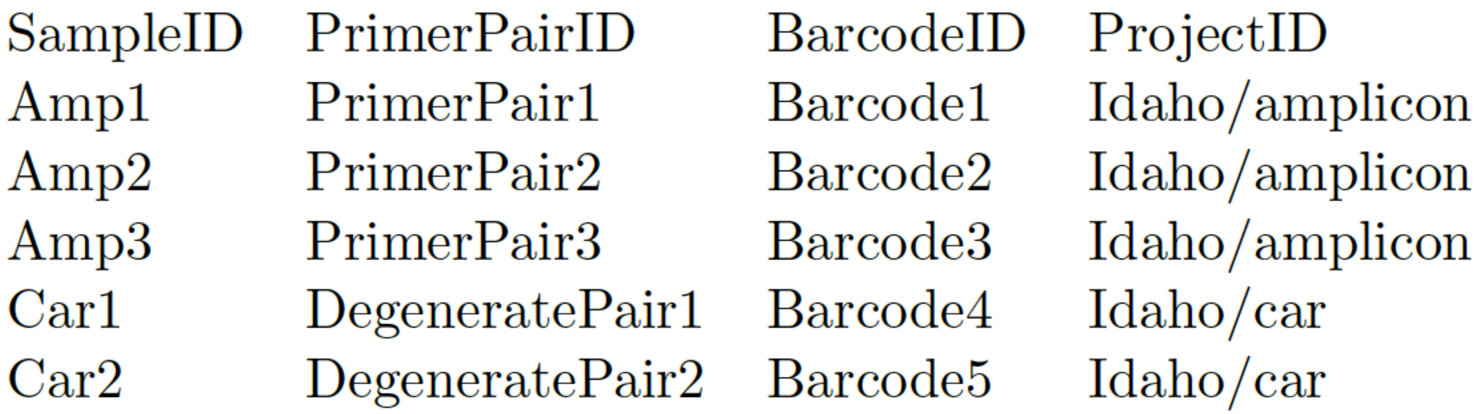


#Read	Pair_ID	Primer_ID	Sequence
P5	PrimerPair1	Primer1Forward	GTAGAGTTTGATCCTGGCTCAG
P5	PrimerPair2	Primer2Forward	CGTAGAGTTTGATCATGGCTCAG
P5	PrimerPair3	Primer3Forward	ACGTAGAGTTTGATTCTGGCTCAG
P5	DegeneratePair1	Degenerate1Forward	GTGARTCATCGAATCTTTG
P5	DegeneratePair2	Degenerate2Forward	CGTGARTCATCGAATCTTTG
P7	PrimerPair1	Primer1Reverse	GTCCTCCGCTTATTGATATGC
P7	PrimerPair2	Primer2Reverse	TGTCCTCCGCTTATTGATATGC
P7	PrimerPair3	Primer3Reverse	ATGTCCTCCGCTTATTGATATGC
P7	DegeneratePair1	Degenerate1Reverse	GGACTACHVGGGTWTCTAAT
P7	DegeneratePair2	Degenerate2Reverse	TGGACTACHVGGGTWTCTAAT

Input Files: Sample Sheet

Requires 4 columns and a header: SampleID samples name, PrimerPairID same as in primer file, barcodeID same as in barcode file, and ProjectID which represents the file prefix for the output and can include a path. SampleID, PrimerPairID, BarcodeID pairs must be unique. In addition for PrimerPairID, can be comma separated, * (match any primer), or '-' should match no primer.

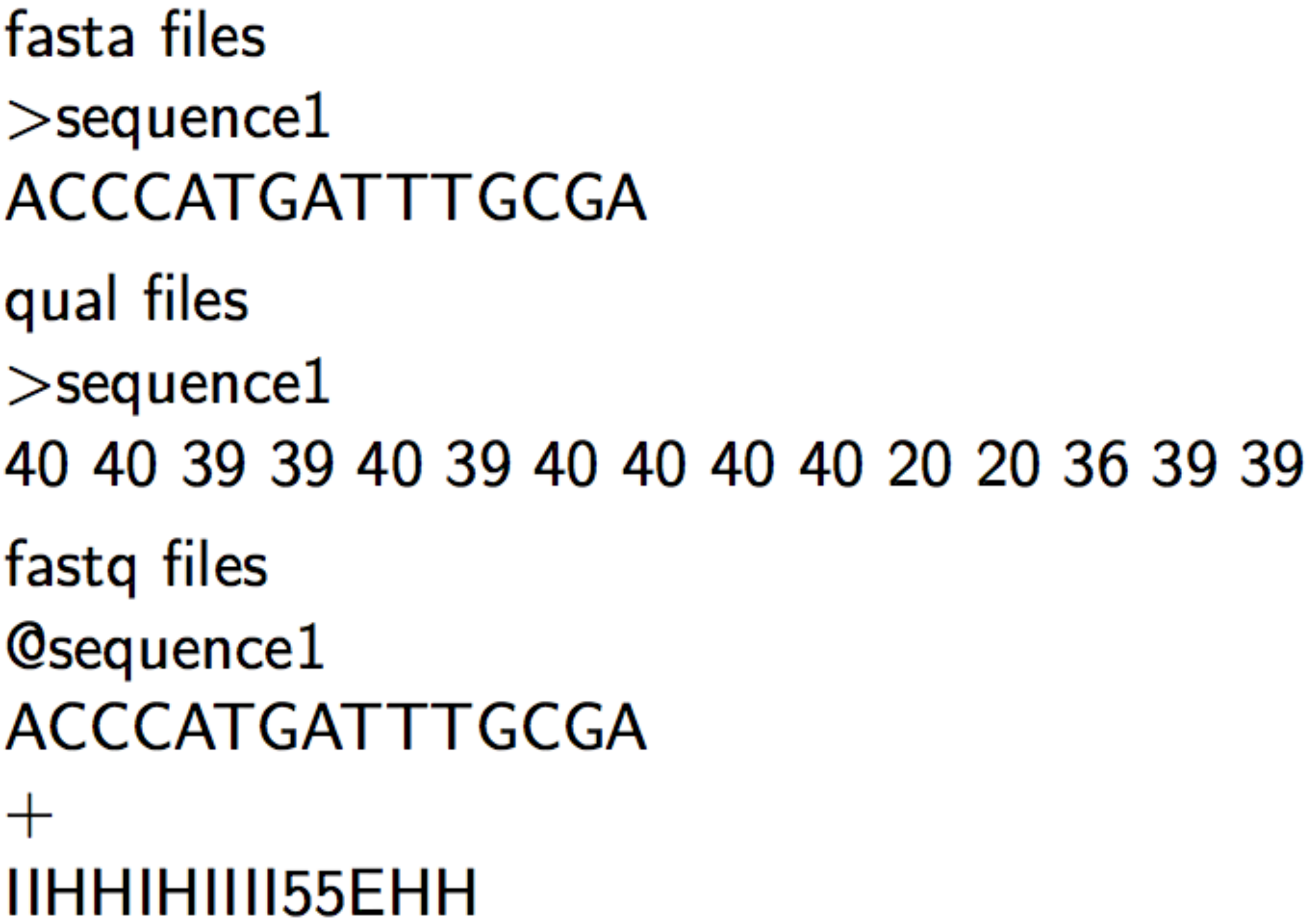
Additional columns are allowed and will be added to the biom file in dbcAmplicons abundances.



SampleID	PrimerPairID	BarcodeID	ProjectID
Amp1	PrimerPair1	Barcode1	Idaho/amplicon
Amp2	PrimerPair2	Barcode2	Idaho/amplicon
Amp3	PrimerPair3	Barcode3	Idaho/amplicon
Car1	DegeneratePair1	Barcode4	Idaho/car
Car2	DegeneratePair2	Barcode5	Idaho/car

Samples
Allowed Characters:
a-zA-Z0-9_-
Projects
Disallowed Characters:
:"\'∗?<>|<space>

Input Files: Sequencing Read files



fasta files

```
>sequence1
ACCCATGATTTGCGA
```

qual files

```
>sequence1
40 40 39 39 40 39 40 40 40 40 20 20 36 39 39
```

fastq files

```
@sequence1
ACCCATGATTTGCGA
+
IIHHIHIIII55EHH
```

Quality Scores

$$Q = -10 log_{10} P$$

Phred Quality Score	Probability of incorrect base call	Base call accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1000	99.9%
40	1 in 10000	99.99%

$Q_{sanger} = -10 log_{10} P$ - based on probability (aka phred)

$Q_{solexa} = -10 log_{10} \frac{P}{1-P}$ - based on odds

S - Sanger	Phred+33,	raw reads typically (0, 40)
X - Solexa	Solexa+64,	raw reads typically (-5, 40)
I - Illumina 1.3+	Phred+64,	raw reads typically (0, 40)
J - Illumina 1.5+	Phred+64,	raw reads typically (3, 40)
L - Illumina 1.8+	Phred+33,	raw reads typically (0, 41)

Illumina Read naming conventions

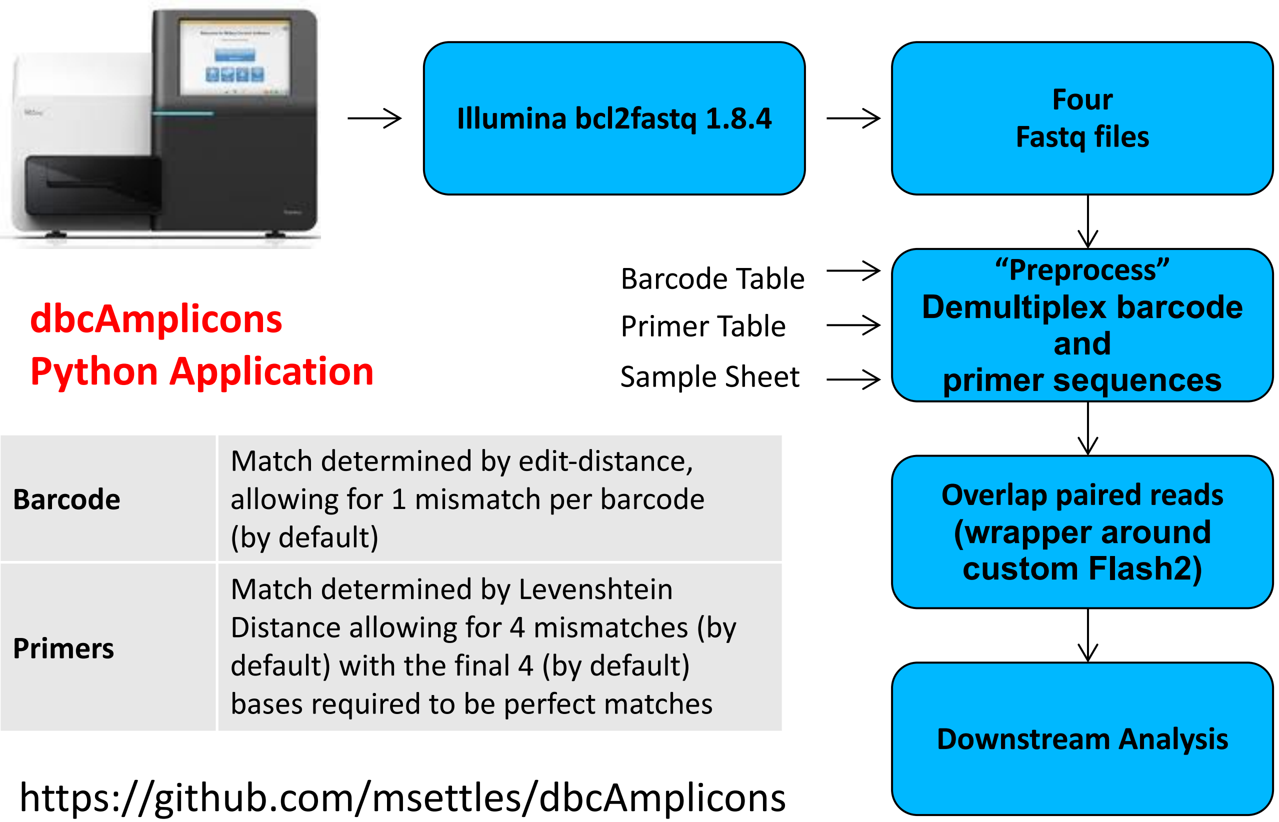
CASAVA 1.8 or greater Read IDs

- @EAS139:136:FC706VJ:2:2104:15343:197393 1:Y:18:ATCACG
  - EAS139 the unique instrument name
  - 136 the run id
  - FC706VJ the flowcell id
  - 2 flowcell lane
  - 2104 tile number within the flowcell lane
  - 15343 ’x’-coordinate of the cluster within the tile
  - 197393 ’y’-coordinate of the cluster within the tile
  - 1 the member of a pair, 1 or 2 (paired-end or mate-pair reads only)
  - Y Y if the read fails filter (read is bad), N otherwise
  - 18 0 when none of the control bits are on, otherwise it is an even number
  - ATCACG index sequence

Reads from the sequencing provider

- Fastq files are actually not raw data from the provider, “raw” data is actually bcl files.
- Sequencing provider will run an application bcl2fastq with a sample sheet to produce demultiplexed (by barcode) fastq files.
- For dbcAmplicons you want to request from your sequencing provider non-demultiplexed fastq (so one set for the entire run) and the index reads.

Bioinformatics


Illumina bcl2fastq 1.8.4
Four Fastq files
Barcode Table
Primer Table
Sample Sheet
"Preprocess" Demultiplex barcode and primer sequences
Overlap paired reads (wrapper around custom Flash2)
Downstream Analysis

dbcAmplicons
Python Application

Barcode	Match determined by edit-distance, allowing for 1 mismatch per barcode (by default)
Primers	Match determined by Levenshtein Distance allowing for 4 mismatches (by default) with the final 4 (by default) bases required to be perfect matches

https://github.com/msettles/dbcAmplicons

Downstream Analysis

Population Community Profiling (i.e. microbial, bacterial, fungal, etc.)

dbcAmplicons Python Application

Screen	Using Bowtie2, screen targets against a reference fasta file, separating reads by those that produce matches and those that do not match sequences in the reference database.
Classify	Wrapper around the MSU Ribosomal Database Project (RDP) Classifier for Bacterial and Archaeal 16S rRNA sequences, Fungal 28S rRNA, fungal ITS regions
Abundance	Reduce RDP classifier results to abundance tables (or biom file format), rows are taxa and columns are samples ready for additional community analysis.

Targeted Re-sequencing

	Consensus - Reduce reads to consensus sequence for each sample and amplicon
R-functions to be added into dbcAmplicons	**Most Common –** Reduce reads to the most commonly occurring read in the sample and amplicon (that is present in at least 5% and 5 reads, by default)
	Haplotypes – Impute the different haplotypes in the sample and amplicon

Supplemental Scripts

- convert2Readto4Read.py
  - For when samples are processed by someone else
- splitReadsBySample.py
  - To facilitate upload to the SRA
- preprocPair_with_inlineBC.py
  - Cut out inline BC and create 4 reads for standard input processing
  - Will work with ”Mills lab” protocol
- dbcVersionReport.sh
  - Print out version numbers of all tools

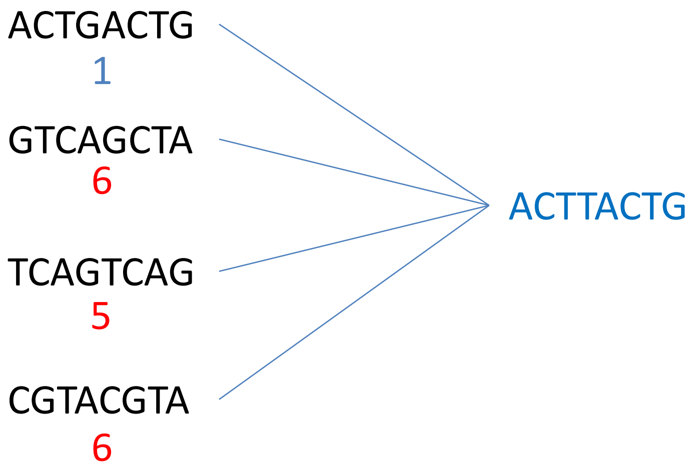
dbcAmplicons: Preprocessing

1. Read in the metadata input tables: Barcodes, Primers (optional), Samples (optional)
2. Read in a batch of reads (default 100,000), for each read
    1. Compare index barcodes to the barcode table, note best matching barcode
    2. Compare 5' end of reads to the primer table, note best matching primer
    3. Compare to barcode:primer pair to the sample table, note sampleID and projectID
    4. If its a legitimate reads (contains matching barcode,primer,sample) output the reads to the output file
3. Output Identified_Barcodes.txt file

Output: Preprocessed reads, Identified_Barcodes.txt file

Barcode/Primer Comparison

Barcode Comparison



ACTGACTG
1

GTCAGCTA
6

TCAGTCAG
5

CGTACGTA
6

ACTTACTG

Compares each barcode to all possible barcodes and returns the best match < desired edit distance

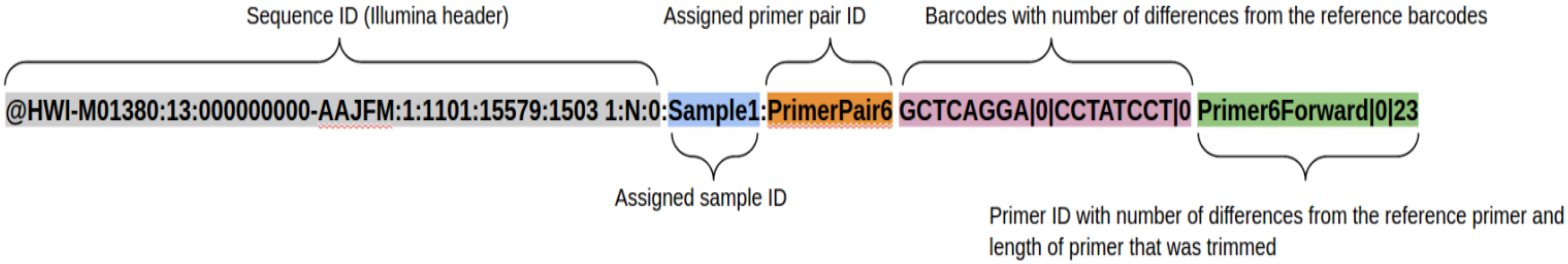
Primer Comparison

GGCTTGGTCATTTAGAGGAAGTAA	Primer 1
TACGGCTTGGTCATTTAGAGGAAGTAA	Primer 2
CGGCTTGGTCATTTAGAGGAAGTAA	Primer 3
ACGGCTTGGTCATTTAGAGGAAGTAA	Primer 4
TACGGACTTG_TCATTTACAGGAAGTAAAAGTCGTAA	Read

Compares the beginning (primer region) of each read to all possible primers and returns the best match < specified maximum Levenshtein disteance + final 4 exact match

The new read header

Header format of all identified sequences:



Sequence ID (Illumina header)

Assigned primer pair ID

Barcodes with number of differences from the reference barcodes

```
@HWI-M01380:13:000000000-AAJFM:1:1101:15579:1503 1:N:0:Sample1:PrimerPair6 GCTCAGGA|0|CCTATCCT|0 Primer6Forward|0|23
```

Assigned sample ID

Primer ID with number of differences from the reference primer and length of primer that was trimmed

```
@HWI-M01380:50:000000000-A641U:1:1101:17127:1556 1:N:0:Sample97:16S GTCGTGAT|0|TAAGTTCC|0 27F_Bif|0|26
GATGAACGCTAGCTACAGGCTTAACACATGCAAGTCGAGGGGCATCAGGAAGAAAGCTTGCTTTCTTTGCTGGCGACCGGCGCACGGGTGAGTAACACGTATC
```

dbcAmplicions: join

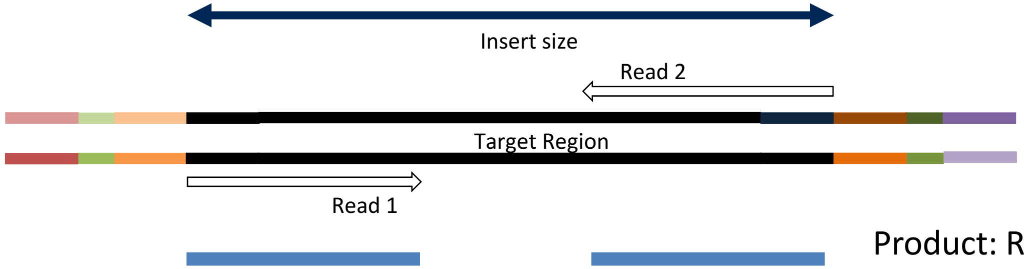
- Uses Flash2 to merge reads that overlap to produce a longer (or sometimes shorter read).
- Modification include:
  - Performs complete overlaps with adapter trimming
  - Allows for different sized reads (after cutting primer off)
  - Discards reads with > 50% Q of 10 or less, which are indicative of adapter/primer dimers

Output:

prefix.notCombined_1.fastq.gz, prefix.notCombined_2.fastq.gz
prefix.extendedFrags.fastq.gz
prefix.hist
prefix.histogram

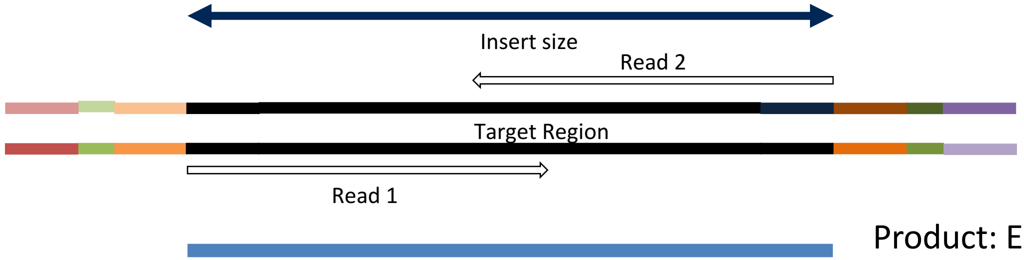
Flash2 – overlapping of reads and adapter removal in paired end reads

Insert size > length of the number of cycles



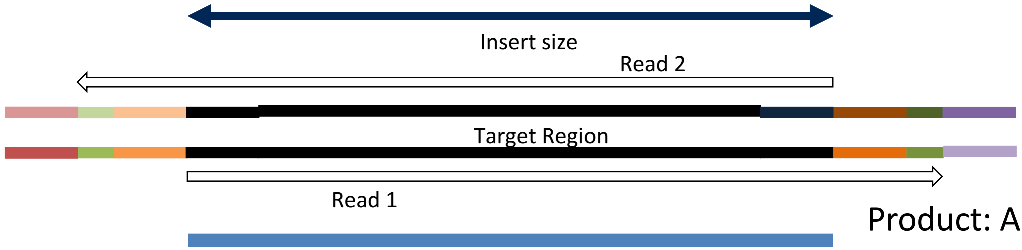
Product: Read Pair

Insert size < length of the number of cycles (10bp min)



Product: Extended, Single

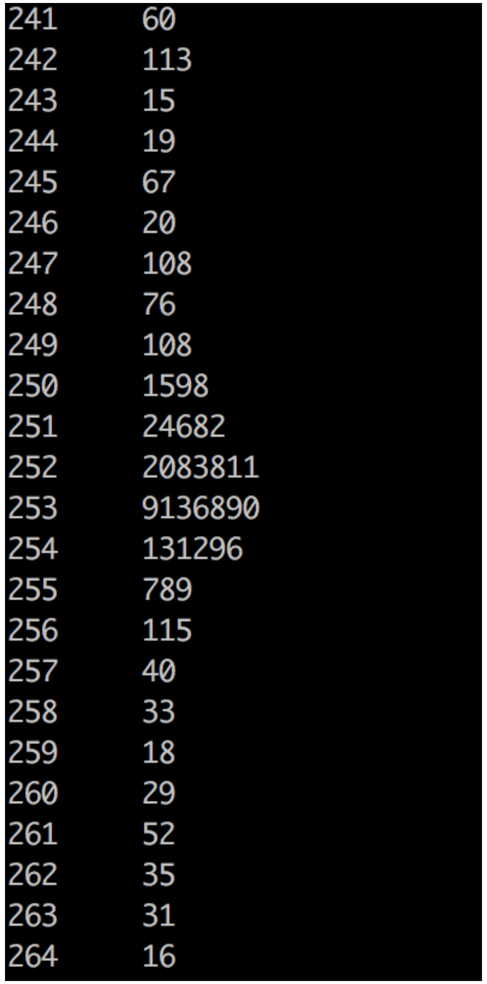
Insert size < length of the read length



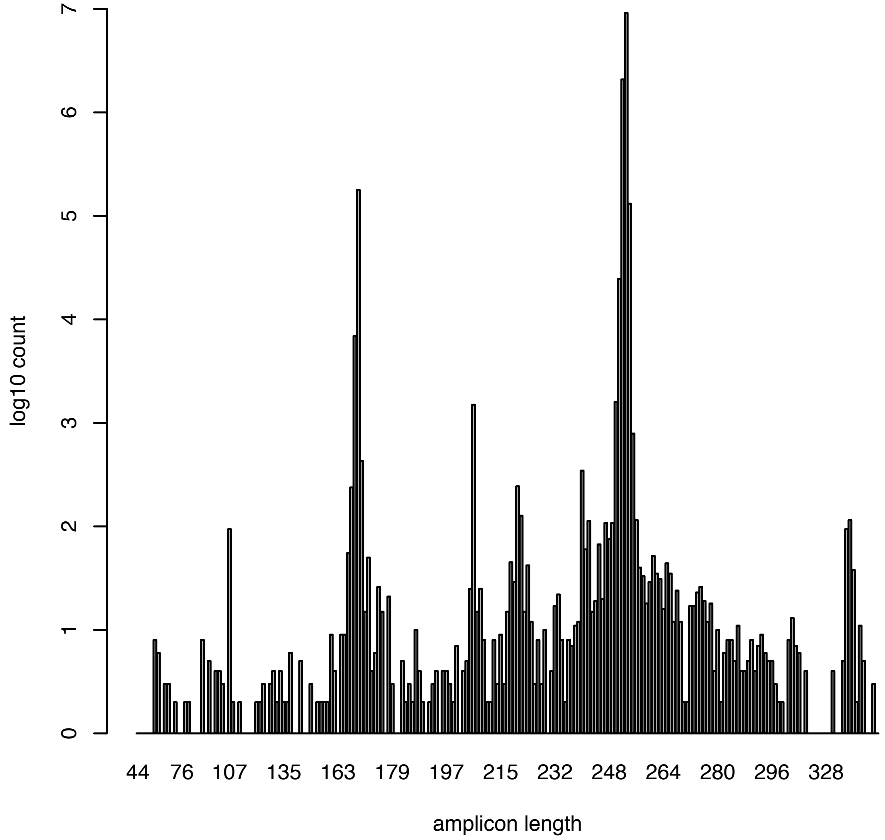
Product: Adapter Trimmed, Single

https://github.com/dstreett/FLASH2

Flash2 typically produces tight sizes



```
241       60
242       113
243       15
244       19
245       67
246       20
247       108
248       76
249       108
250       1598
251       24682
252       2083811
253       9136890
254       131296
255       789
256       115
257       40
258       33
259       18
260       29
261       52
262       35
263       31
264       16
```



dbcAmplicons: classify

- Uses the RDP (Ribosomal Database Project) classifier for bacterial and archaeal 16S, fungal LSU, ITS warcup/unite databases. You can provide your own training database

- Classifies sequences to the closest taxonomic reference provides a bootstrap score for reliability

- Concatenates Paired-end reads
  - Can trim off low quality ends, to some value Q

Output: fixrank file

```
HWI-M01380:26:000000000-ABHNY:1:2116:24606:7147|Slashpile27:16sV1V3:470    Bacteria    domain    1.0    "Proteobacteria“
phylum    1.0 ………. Through Genus/Species
```

Direct Classification - RDP

- Ribosomal Database Project (RDP) - naïve Bayesian Classifier
  - Compares each read to a database
    - Database is updates periodically
  - Compares by k-mers (15 mers)
  - 100 bootstraps to establish confidence in result

- Order does not matter, no 3% !

- Drawbacks
  - Accepts only fasta (though website implies fastq) files
  - Can be slow
  - Down to genus only (for 16s, species for ITS)
  - Kmer database are based on whole 16s
  - Cannot group together unknown OTUs that represent unique taxa

Clustering

- Clustering – “Because of the increasing sizes of today’s amplicon datasets, fast and greedy *de novo* clustering heuristics are the preferred and only practical approach to produce OTUs”. I DISAGREE

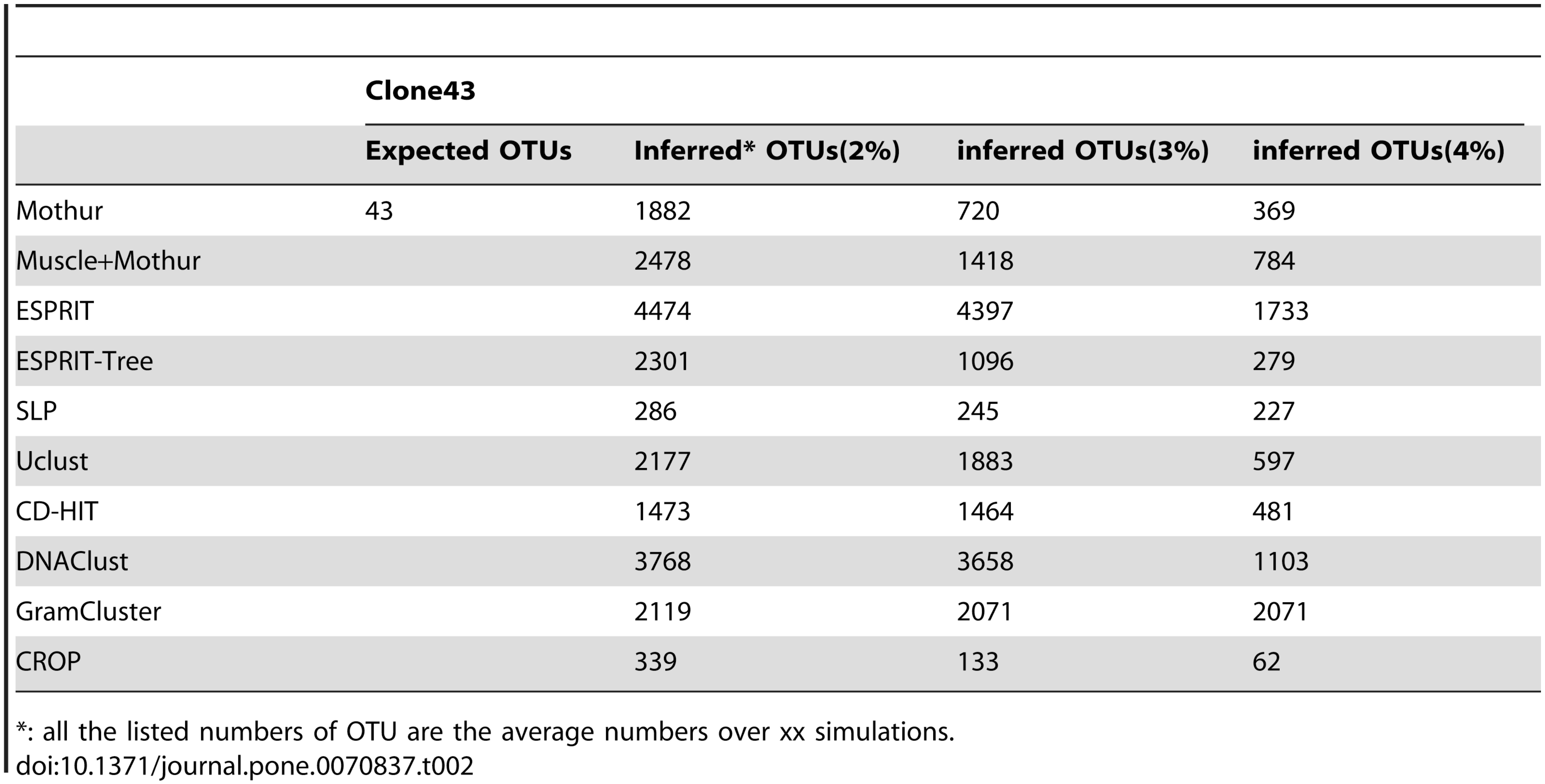
Shared steps in these current algorithms are:

1. An amplicon is drawn out of the amplicon pool and becomes the center of a new OTU (centroid selection)
2. This centroid is then compared to all other amplicons remaining in the pool.
3. Amplicons for which the distance is within a global clustering threshold, *t (e.g. 3%),* to the centroid are moved from the pool to the OUT
4. The OTU is then closed. These steps are repeated as long as amplicons remain in the pool.

Reasons why I’m not a fan

1. Little to no biological rational to any of the clustering parameters, modify the parameters to get a result you like.
2. Dependent on ordering, reorder our reads you can get different set of OTUs. Often not repeatable from run to run.
3. 3% (or any other cutoff) is BS.
4. Most clustering algorithms do not consider sequencing errors.
5. If you generate more data you have to start the clustering process all over again as population of sequences matters.
6. I’m sure there is more

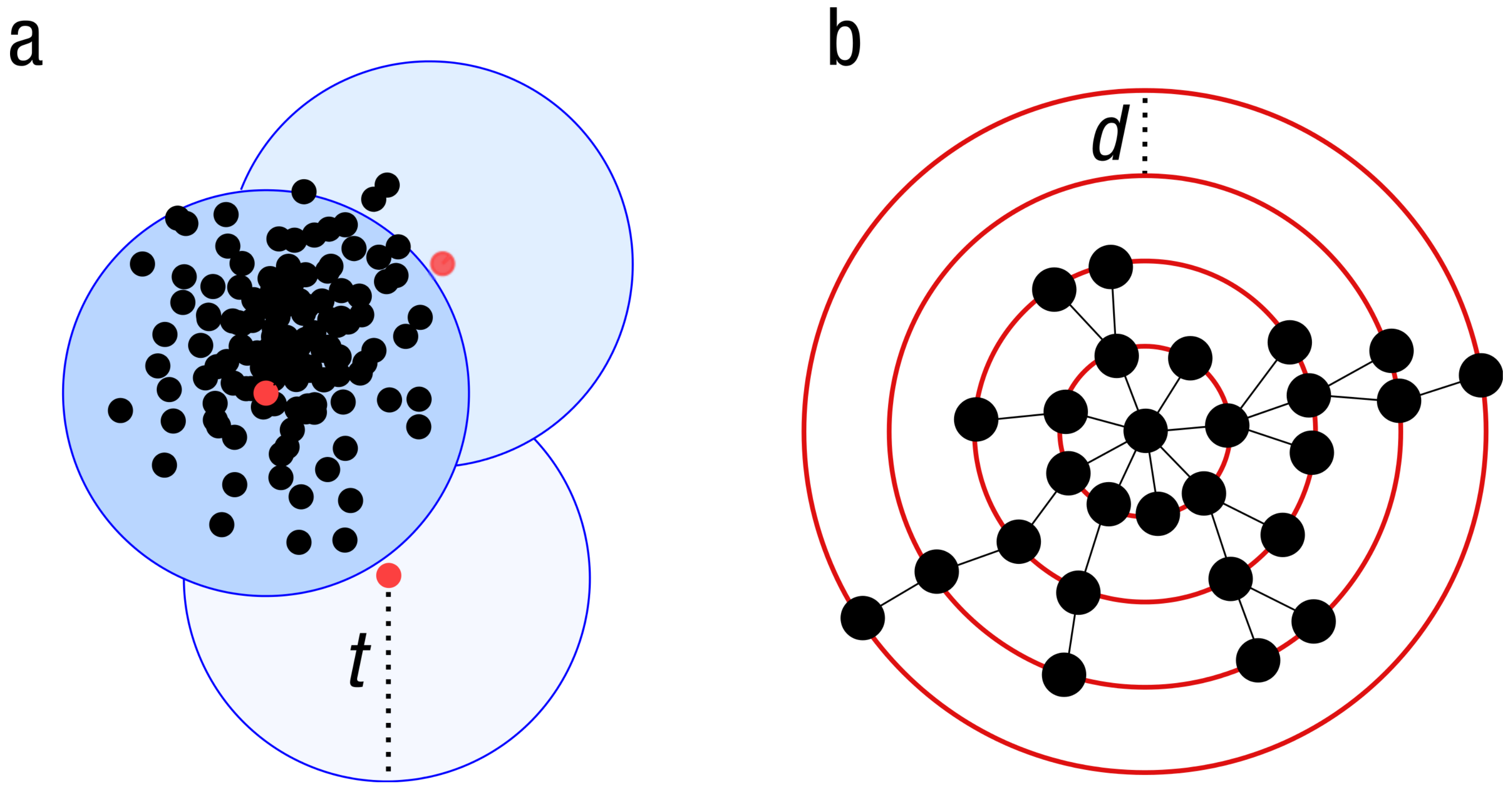
OTU clustering Comparison

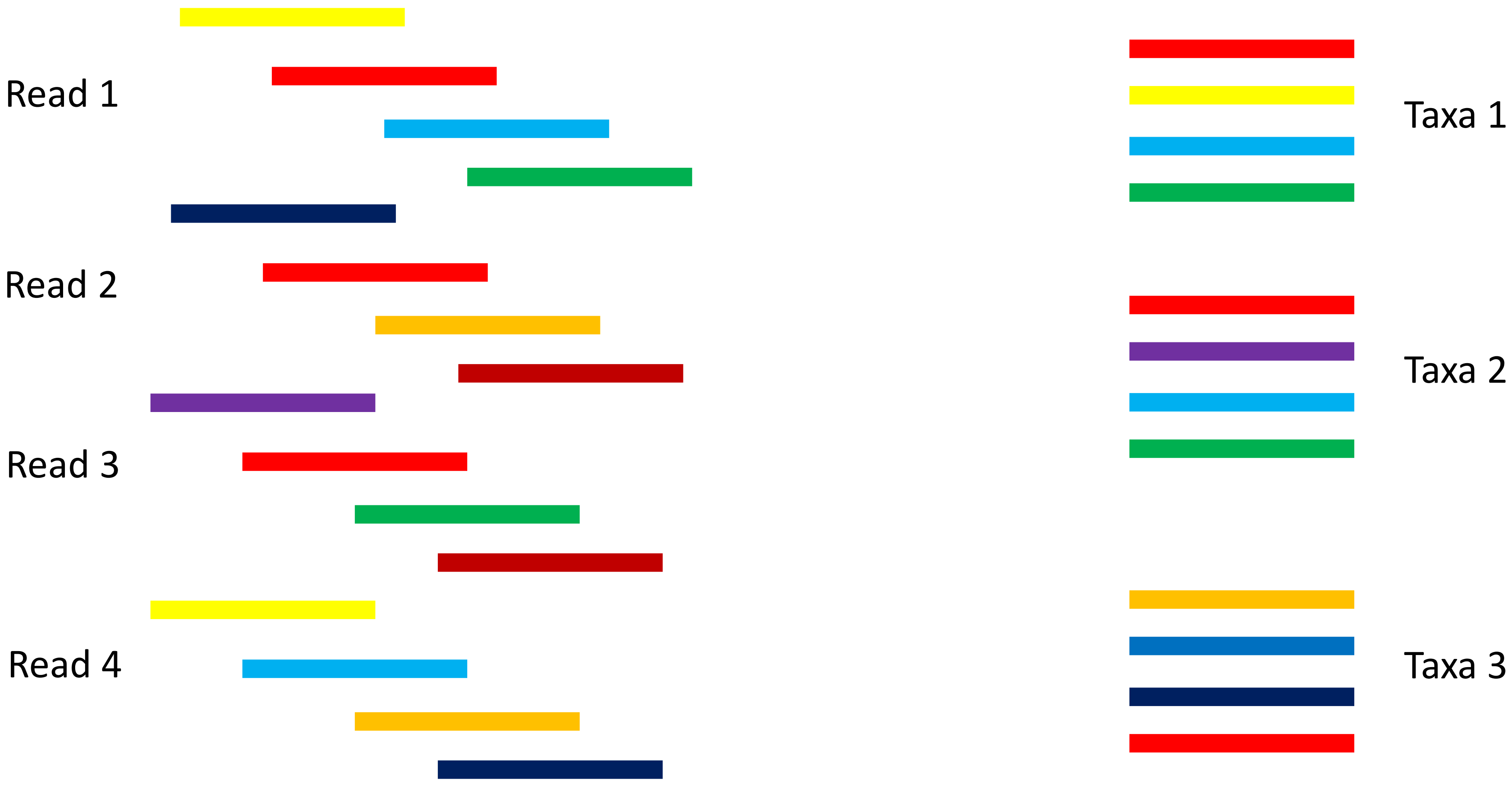


	Clone43			
	Expected OTUs	Inferred* OTUs(2%)	inferred OTUs(3%)	inferred OTUs(4%)
Mothur	43	1882	720	369
Muscle+Mothur		2478	1418	784
ESPRIT		4474	4397	1733
ESPRIT-Tree		2301	1096	279
SLP		286	245	227
Uclust		2177	1883	597
CD-HIT		1473	1464	481
DNAClust		3768	3658	1103
GramCluster		2119	2071	2071
CROP		339	133	62

*: all the listed numbers of OTU are the average numbers over xx simulations.
doi:10.1371/journal.pone.0070837.t002

Chen W, Zhang CK, Cheng Y, Zhang S, Zhao H (2013) A Comparison of Methods for Clustering 16S rRNA Sequences into OTUs. PLoS ONE 8(8): e70837. doi:10.1371/journal.pone.0070837
http://journals.plos.org/plosone/article?id=info:doi/10.1371/journal.pone.0070837


a
t
b
d



Read 1

Read 2

Read 3

Read 4

Taxa 1

Taxa 2

Taxa 3

dbcAmplicons: abundance

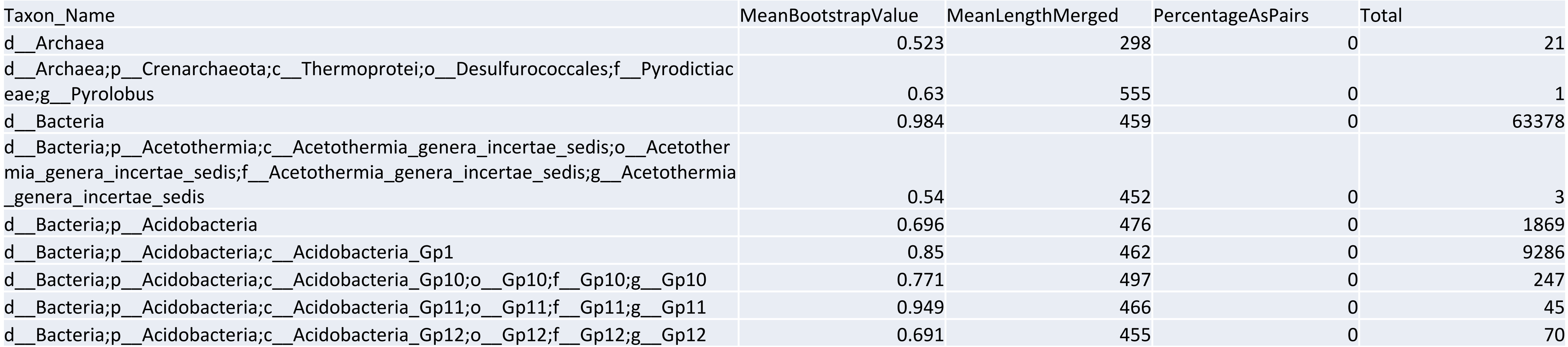
- Takes fixrank file(s) outputs abundances tables and taxa_info table
- Abundance tables
  - Rows are taxa
  - Columns are samples
  - Counts of the number of amplicons for each taxa/samples
- Proportions tables
  - Same as abundance but each cell is the proportion of amplicons (so counts in cell divided by the columns sum)
- Biom file (Biological Observation Matrix)
  - JSON file format for microbiome files
  - http://biom-format.org
  - Abundance tables are 0 heavy, a biom file removes the 0's as well as stores extra metadata

Abundance tables and Biom files

Taxon_Name	Level	Slashpile1	Slashpile10	Slashpile11	Slashpile13	Slashpile14	Slashpile15	Slashpile16	Slashpile17
Archaea	domain	0	0	0	1	1	1	1	0
Pyrolobus	genus	1	0	0	0	0	0	0	0
Bacteria	domain	2981	1479	110	2674	1732	2707	1303	2706
Acetothermia_genera_incertae_sedis	genus	0	0	0	0	0	0	0	1
Acidobacteria	phylum	84	34	4	85	60	110	17	60
Acidobacteria_Gp1	class	376	252	31	444	378	565	13	218
Gp10	genus	17	5	0	1	6	5	2	0
Gp11	genus	3	0	0	4	0	12	0	0
Gp12	genus	6	0	0	1	2	2	0	1
Gp13	genus	19	1	0	5	4	11	1	4
Gp15	genus	47	6	0	10	3	29	1	27
Gp16	genus	187	135	6	132	105	171	12	69
Gp17	genus	21	6	1	12	8	11	15	6
Gp18	genus	0	0	0	0	0	1	0	0
Gp19	genus	1	0	0	0	1	0	0	1
Acidicapsa	genus	9	17	6	29	11	11	0	1
Acidipila	genus	17	9	8	18	20	36	0	7
Acidobacterium	genus	13	4	0	4	2	5	1	0
Bryocella	genus	75	75	8	104	115	166	1	125

The **BIOM file format** (canonically pronounced *biome*) is designed to be a general-use format for representing biological sample by observation contingency tables. BIOM is a recognized standard for the **Earth Microbiome Project** and is a **Genomics Standards Consortium** supported project. Contains the abundance counts, the sample names, full taxonomic string [domain through genus/species], and any sample metadata in the sample sheet.

Samples.taxa_info.txt



Taxon_Name	MeanBootstrapValue	MeanLengthMerged	PercentageAsPairs	Total
d__Archaea	0.523	298	0	21
d__Archaea;p__Crenarchaeota;c__Thermoprotei;o__Desulfurococcales;f__Pyrodictiaceae;g__Pyrolobus	0.63	555	0	1
d__Bacteria	0.984	459	0	63378
d__Bacteria;p__Acetothermia;c__Acetothermia_genera_incertae_sedis;o__Acetothermia_genera_incertae_sedis;f__Acetothermia_genera_incertae_sedis;g__Acetothermia_genera_incertae_sedis	0.54	452	0	3
d__Bacteria;p__Acidobacteria	0.696	476	0	1869
d__Bacteria;p__Acidobacteria;c__Acidobacteria_Gp1	0.85	462	0	9286
d__Bacteria;p__Acidobacteria;c__Acidobacteria_Gp10;o__Gp10;f__Gp10;g__Gp10	0.771	497	0	247
d__Bacteria;p__Acidobacteria;c__Acidobacteria_Gp11;o__Gp11;f__Gp11;g__Gp11	0.949	466	0	45
d__Bacteria;p__Acidobacteria;c__Acidobacteria_Gp12;o__Gp12;f__Gp12;g__Gp12	0.691	455	0	70

Supplies extra information about the tax identified in the experiment as well as the full taxonomic path.

Future Directions

- dbcAmplicons is a data reduction pipeline, produces abundance/biome files, post processing most typically done in R.
- Include “error-correcting barcodes” in demultiplexing
- Identification of PCR duplicates (using UMI)
- Replace RDP classification with another scheme
  - Have ideas (for years) but no time
- Use amplicon length in classification
- Include screening of diversity sample in preprocessing to get an idea of actual proportion in the pool
- Incorporate the R genotyping pipeline into dbcAmplicons
  - Extend to inferring copy number (or ploidy levels)
- Correct for copy number (16s)
- Output data for rarefaction curves

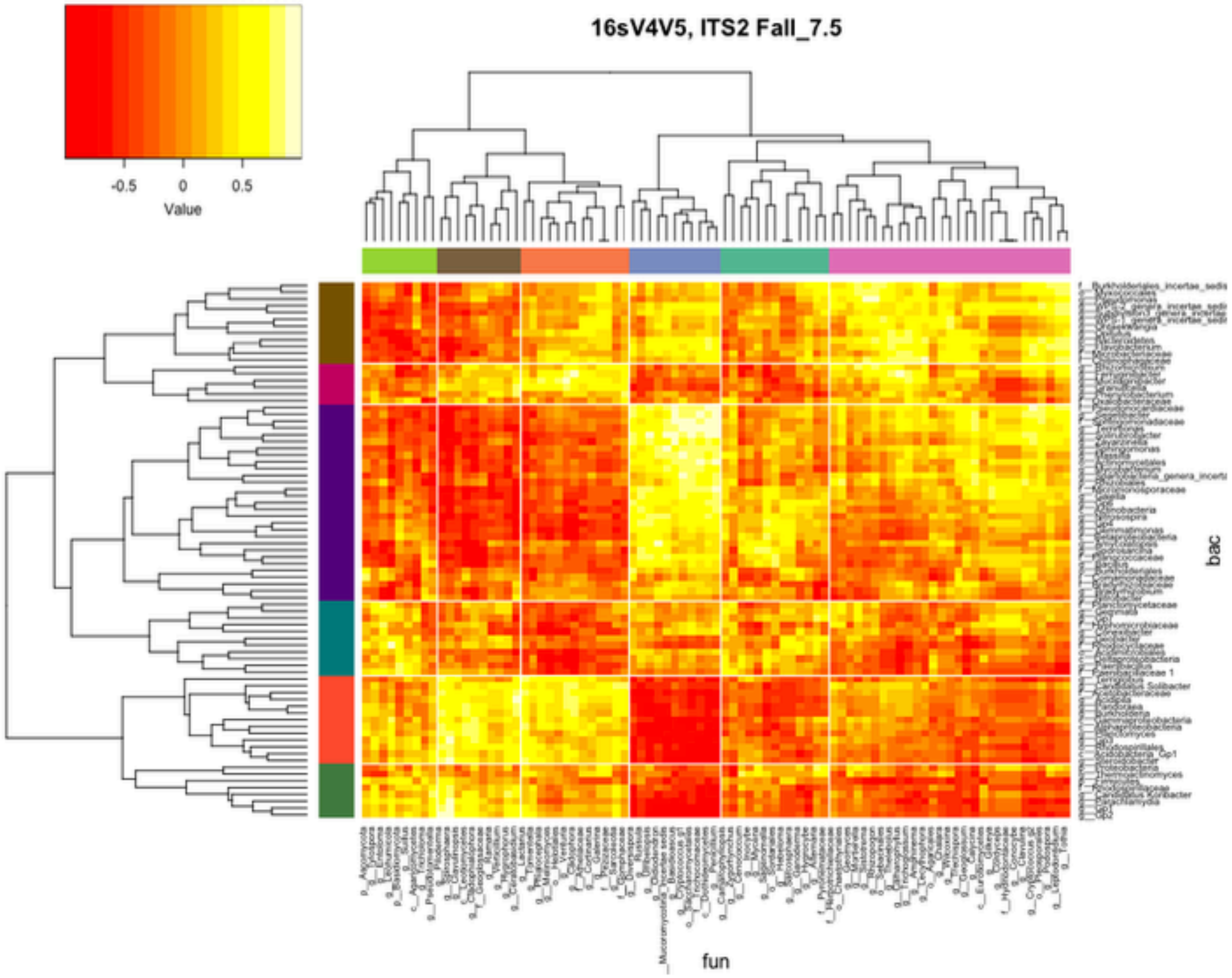
Post Processing

- I pretty much do all of my post analysis (abundance table, Biome) in R
  - Common Packages
    - Vegan
      - ❖ https://github.com/vegandevs/vegan
    - Vegetarian
      - ❖ https://github.com/cran/vegetarian
    - Phyloseq (uses vegan, ade4, ape, picante)
      - ❖ https://joey711.github.io/phyloseq/
- Ecological Diversity Analysis
  - how does the structure of OTU across samples/groups compare
- Ordination Analysis (multivariate analysis)
  - Visualize the relative similarity/dissimilarity across samples, test for taxa/environment relationships
- Differential Abundance Analysis (univariate analysis)
  - Uses tools from RNAseq (limma, edgeR)
- Visualization (temporal, heatmaps, ’trees’, more)

Standardization/Normalization

- Relative (proportional) abundances
  - Divide by sum of sample, values 0-100%
- LogCPM from RNAseq
- Hellinger standardization
  - http://biol09.biol.umontreal.ca/PLcourses/Section_7.7_Transformations.pdf
- Others
  - Wisconsin

Multi community analysis


16sV4V5, ITS2 Fall_7.5
Value
bac
fun