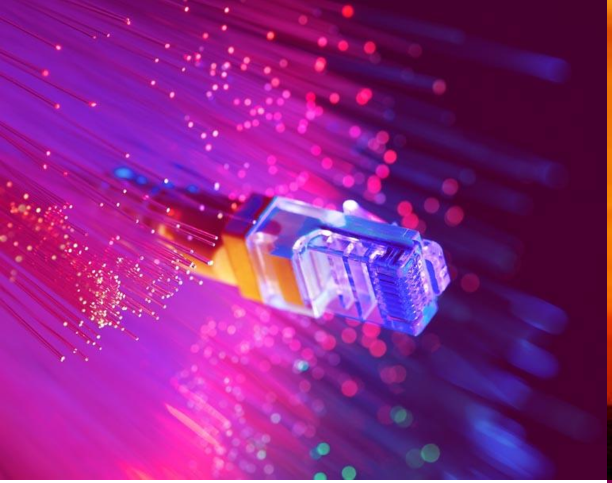


Microbial community analysis with Chipster

20.-21.5.2021

Eija Korpelainen, Jesse Harrison

CSC – Suomalainen tutkimuksen, koulutuksen, kulttuurin ja julkishallinnon ICT-osaamiskeskus

What will I learn?

- Microbial community analysis of amplicon sequencing data
  - Central concepts
  - Analysis steps
  - File formats
- MiSeq 16S data is used in the exercises, but we discuss also how to analyze
  - IonTorrent data
  - ITS data
- How to operate the Chipster software

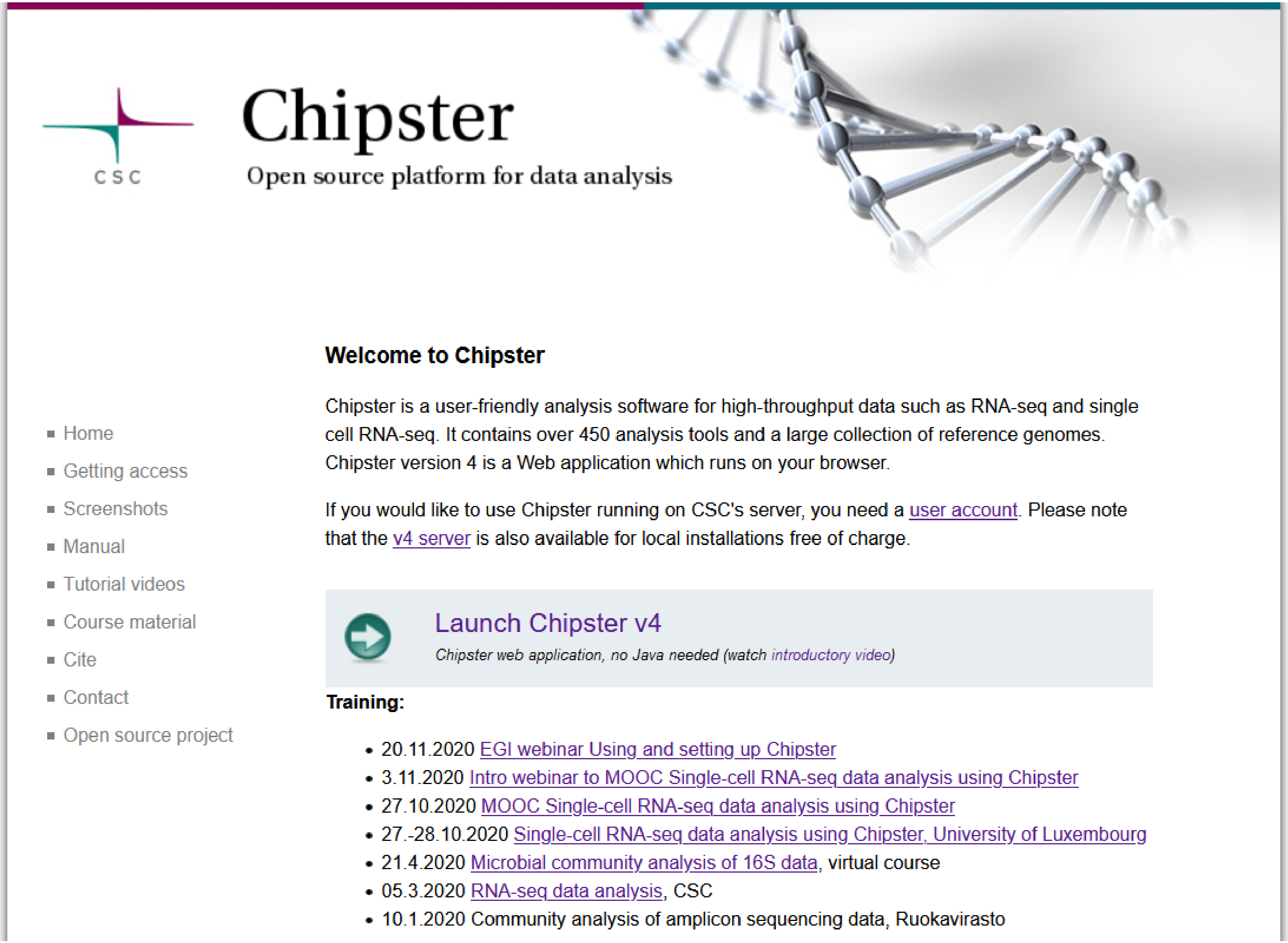
Understanding data analysis - why?

- Bioinformaticians might not always be available when needed
- Biologists know their own experiments best
  - Potential batch effects etc
- Allows you to design experiments better → less money wasted
- Allows you to discuss more easily with bioinformaticians

Introduction to Chipster

- User-friendly analysis software for high-throughput data
- Provides an easy access to over 450 analysis tools
  - Command line tools
  - R/Bioconductor packages
- Free, open source software
- What can I do with Chipster?
  - analyze high-throughput data
  - visualize data efficiently
  - share analysis sessions

Chipster website (https://chipster.csc.fi/)



Chipster

Open source platform for data analysis

Welcome to Chipster

Chipster is a user-friendly analysis software for high-throughput data such as RNA-seq and single cell RNA-seq. It contains over 450 analysis tools and a large collection of reference genomes. Chipster version 4 is a Web application which runs on your browser.

If you would like to use Chipster running on CSC's server, you need a user account. Please note that the v4 server is also available for local installations free of charge.

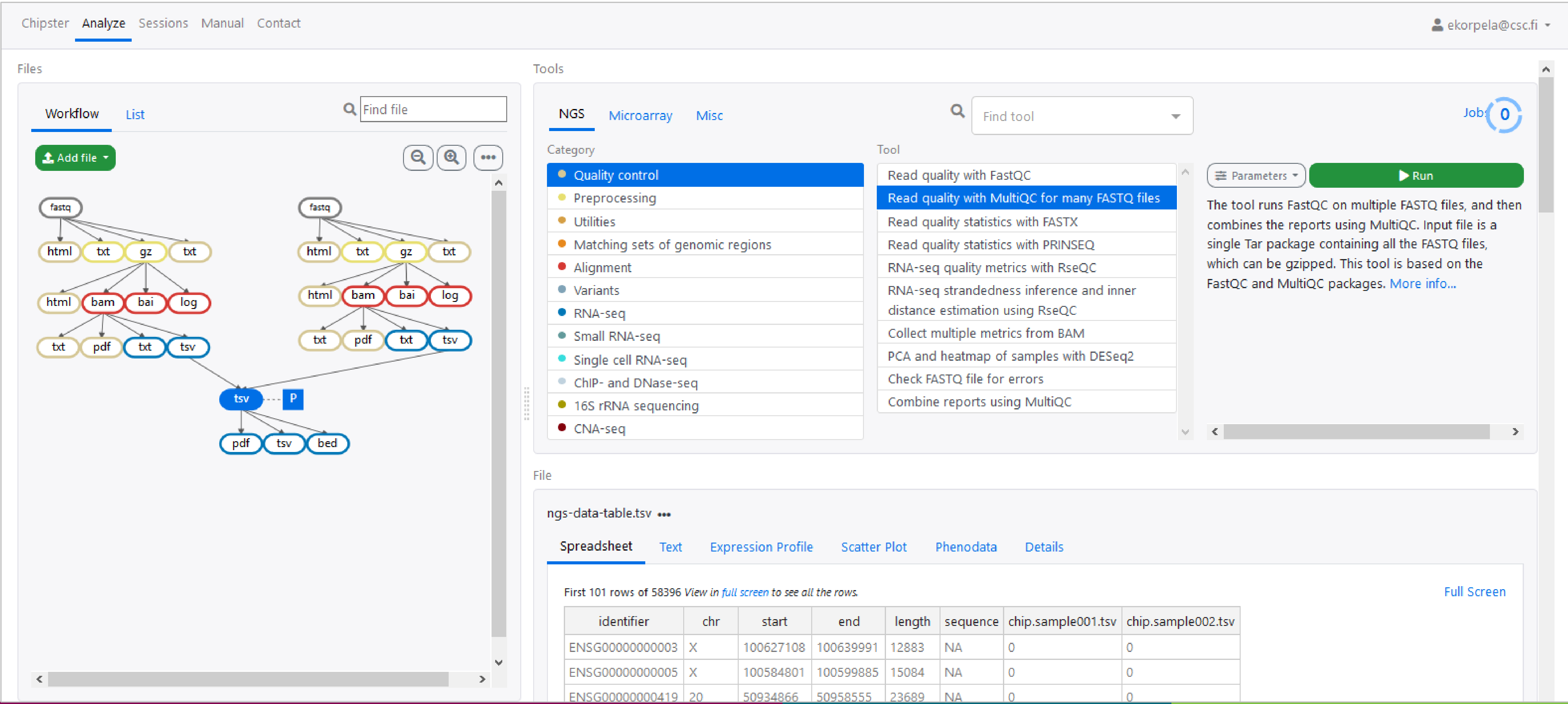
Launch Chipster v4

Chipster web application, no Java needed (watch introductory video)

Training:

- 20.11.2020 EGI webinar Using and setting up Chipster
- 3.11.2020 Intro webinar to MOOC Single-cell RNA-seq data analysis using Chipster
- 27.10.2020 MOOC Single-cell RNA-seq data analysis using Chipster
- 27.-28.10.2020 Single-cell RNA-seq data analysis using Chipster, University of Luxembourg
- 21.4.2020 Microbial community analysis of 16S data, virtual course
- 05.3.2020 RNA-seq data analysis, CSC
- 10.1.2020 Community analysis of amplicon sequencing data, Ruokavirasto

Chipster user interface



Category

- Quality control
- Preprocessing
- Utilities
- Matching sets of genomic regions
- Alignment
- Variants
- RNA-seq
- Small RNA-seq
- Single cell RNA-seq
- ChIP- and DNase-seq
- 16S rRNA sequencing
- CNA-seq

Tool

- Read quality with FastQC
- Read quality with MultiQC for many FASTQ files
- Read quality statistics with FASTX
- Read quality statistics with PRINSEQ
- RNA-seq quality metrics with RseQC
- RNA-seq strandedness inference and inner distance estimation using RseQC
- Collect multiple metrics from BAM
- PCA and heatmap of samples with DESeq2
- Check FASTQ file for errors
- Combine reports using MultiQC

The tool runs FastQC on multiple FASTQ files, and then combines the reports using MultiQC. Input file is a single Tar package containing all the FASTQ files, which can be gzipped. This tool is based on the FastQC and MultiQC packages. More info...

ngs-data-table.tsv •••

First 101 rows of 58396 *View in full screen to see all the rows.*

identifier	chr	start	end	length	sequence	chip.sample001.tsv	chip.sample002.tsv
ENSG00000000003	X	100627108	100639991	12883	NA	0	0
ENSG00000000005	X	100584801	100599885	15084	NA	0	0
ENSG00000000419	20	50934866	50958555	23689	NA	0	0

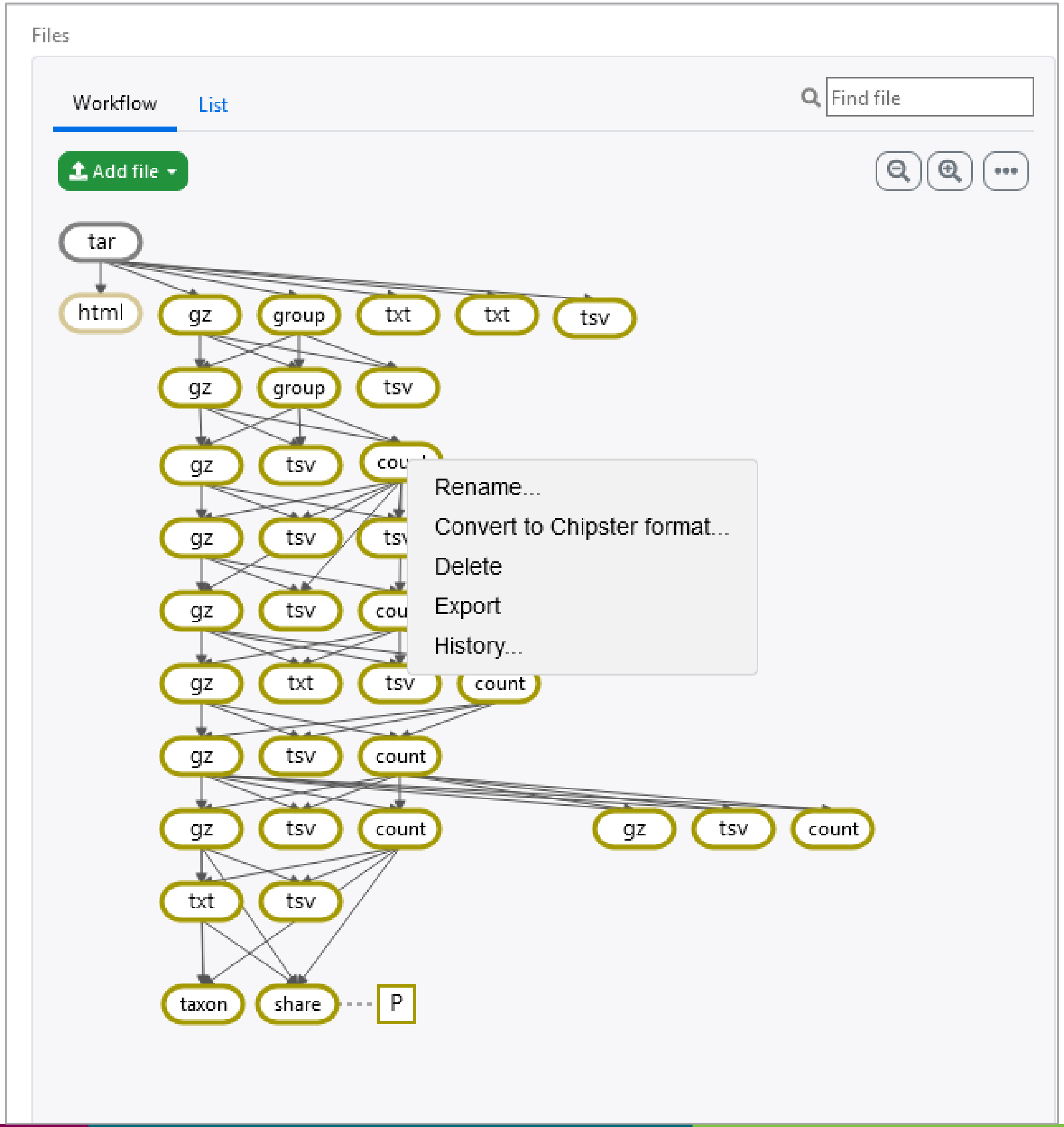
Analysis sessions

- Your analysis is saved automatically in the cloud
  - Session includes all the files, their relationships and metadata (what tool and parameters were used to produce each file).
  - Session is a single .zip file.
  - Note that cloud sessions are not stored forever! Remember to download the session when ready.
- You can share sessions with other Chipster users
  - You can give either read-only or read-write access
- If your analysis job takes a long time, you don’t need to keep Chipster open:
  - Wait that the data transfer to the server has completed (job status = running)
  - Close Chipster
  - Open Chipster later and the results will be there

Workflow view

- Shows the relationships of the files
- You can move the boxes around
- Several files can be selected by
  - keeping the Ctrl key down
  - drawing a box around them

- Right click allows you to
  - download a file ("Export")
  - delete a file
  - view analysis history



Options for importing data to Chipster

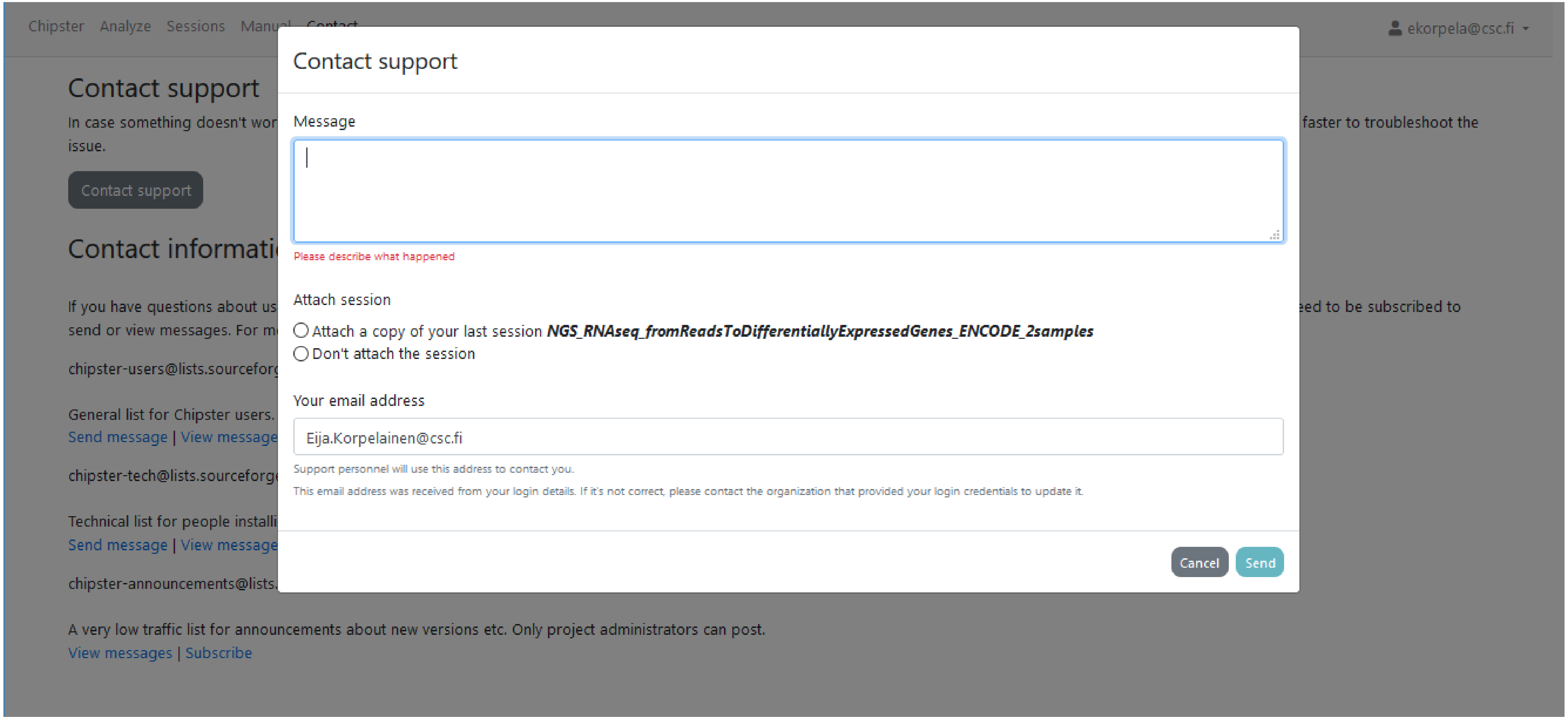
- Add file button
  - Upload files
  - Upload folder
  - Download from URL

- Sessions tab
  - Import session file

- Tools
  - Import from Illumina BaseSpace
    - Utilites / Retrieve data from Illumina BaseSpace
    - Access token needed
  - Import from SRA database
    - Utilities / Retrieve FASTQ or BAM files from SRA
  - Import from Ensembl database
    - Utilities / Retrieve data for a given organism in Ensembl
  - Import from URL
    - Utilities / Download file from URL directly to server

Problems? Send us a support request

-request includes the error message and (optionally) a link to your session



Contact support

In case something doesn't wor... faster to troubleshoot the issue.

Contact informati...

If you have questions about us... eed to be subscribed to send or view messages. For m...

chipster-users@lists.sourceforg...

General list for Chipster users.
Send message | View message...

chipster-tech@lists.sourceforge...

Technical list for people installi...
Send message | View message...

chipster-announcements@lists....

A very low traffic list for announcements about new versions etc. Only project administrators can post.
View messages | Subscribe

Contact support

Message

Please describe what happened

Attach session

○ Attach a copy of your last session ***NGS_RNAseq_fromReadsToDifferentiallyExpressedGenes_ENCODE_2samples***
○ Don't attach the session

Your email address

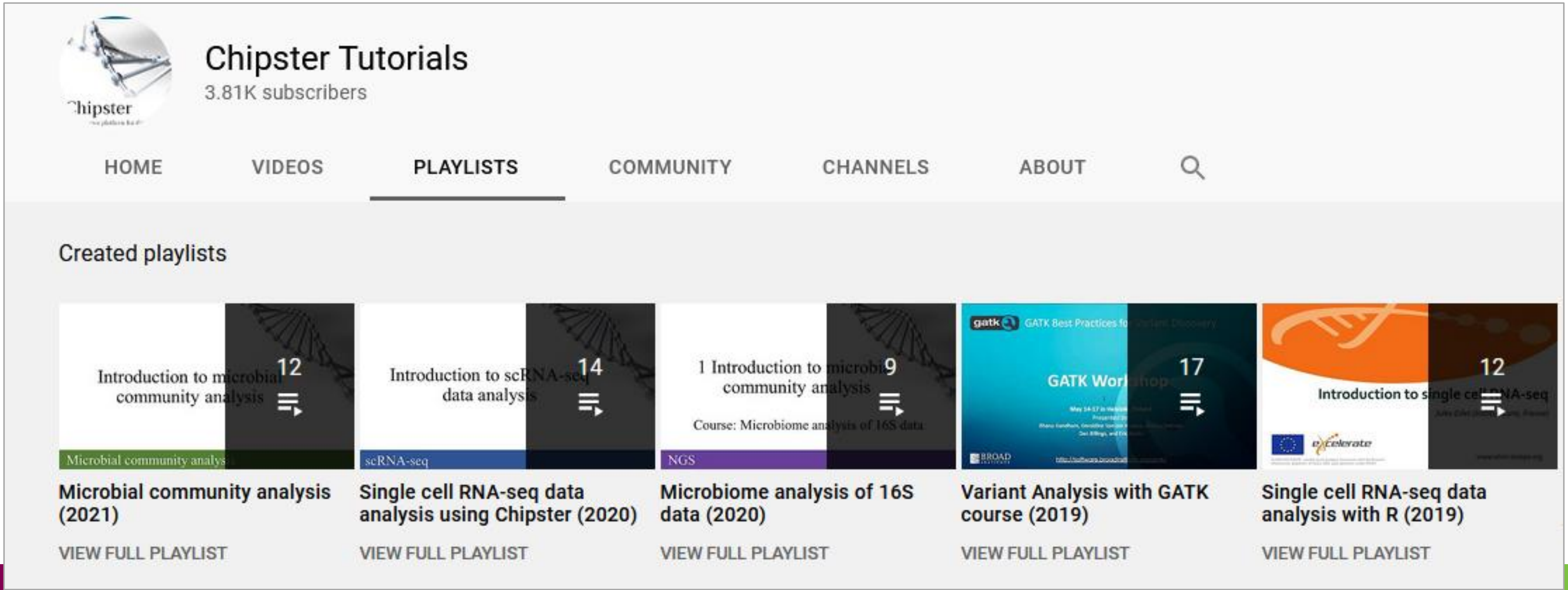
Eija.Korpelainen@csc.fi

Support personnel will use this address to contact you.

This email address was received from your login details. If it's not correct, please contact the organization that provided your login credentials to update it.

More info

- chipster@csc.fi
- http://chipster.csc.fi
- Chipster tutorials in YouTube
- https://chipster.csc.fi/manual/courses.html



Chipster Tutorials
3.81K subscribers

Created playlists

Microbial community analysis (2021)
VIEW FULL PLAYLIST

Single cell RNA-seq data analysis using Chipster (2020)
VIEW FULL PLAYLIST

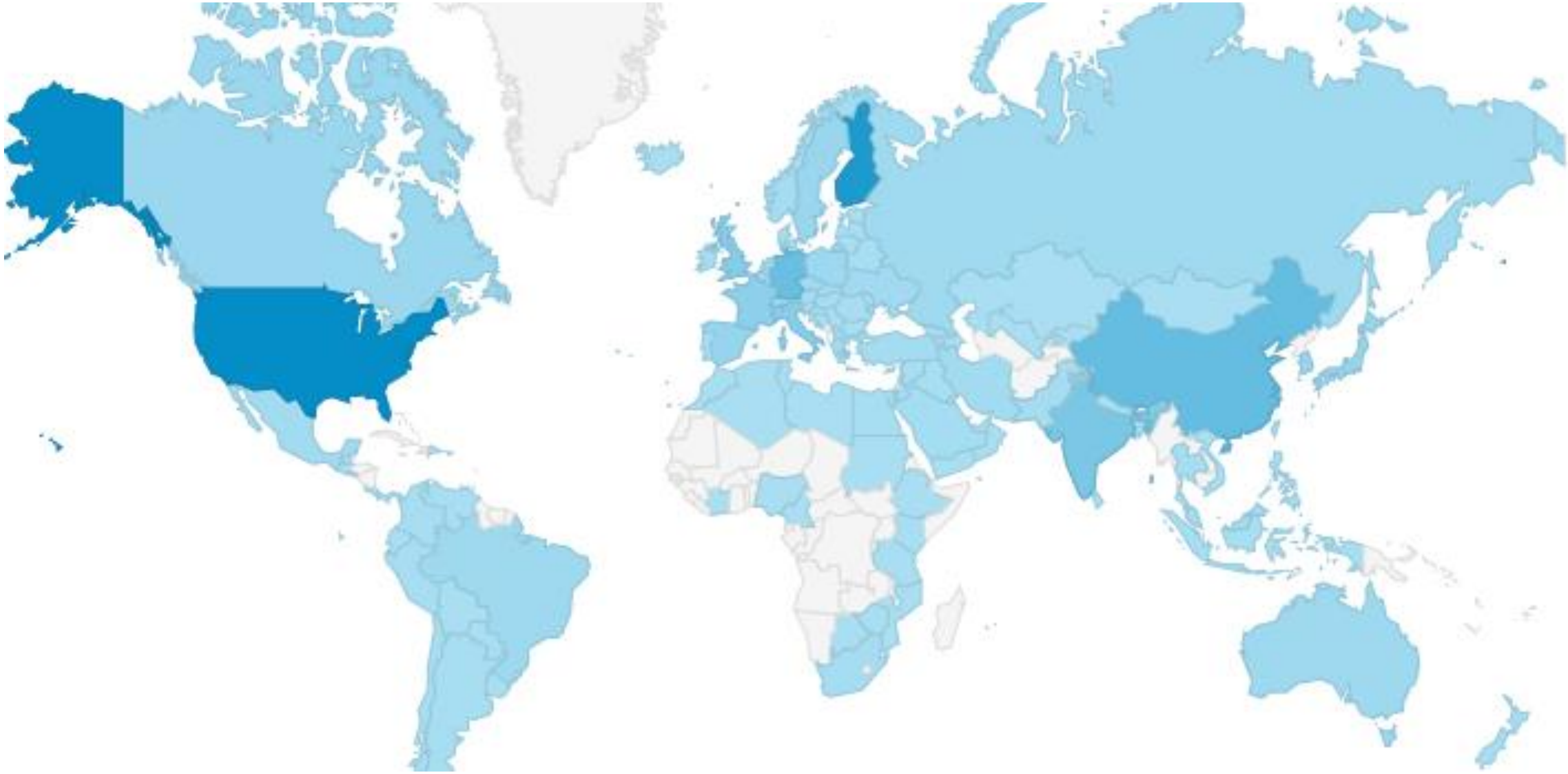
Microbiome analysis of 16S data (2020)
VIEW FULL PLAYLIST

Variant Analysis with GATK course (2019)
VIEW FULL PLAYLIST

Single cell RNA-seq data analysis with R (2019)
VIEW FULL PLAYLIST

Acknowledgements to Chipster users and contibutors

Users' feedback and ideas have helped us to shape the software over the years.
Let us know what needs to be improved!



Introduction to microbial community analysis

Outline

- What questions does it answer
- How is it done
- What are the main steps

Microbial community analysis

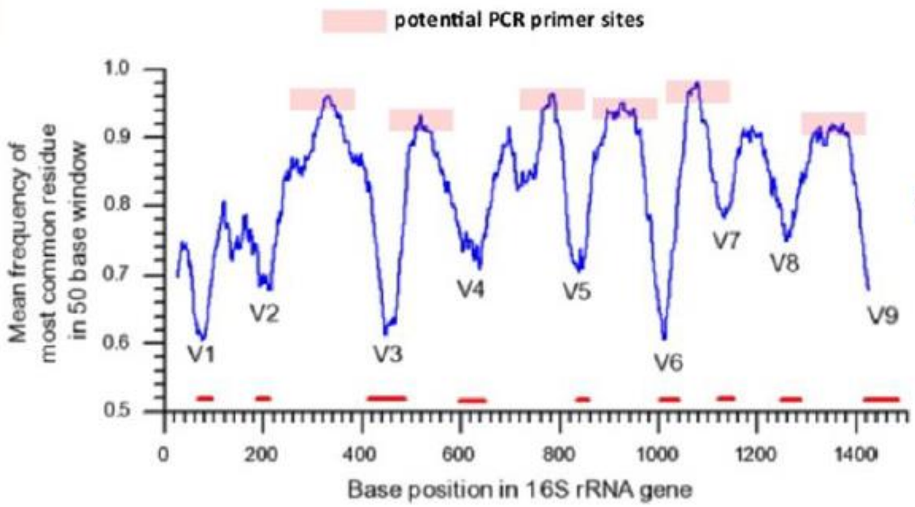
- Answers the questions who are there and in what proportions if compared to your other samples
  - It will not confirm that someone isn't there (sampling depth, primer/sequencing bias)
- Specific primers are used to amplify a region of one gene
  - Bacterial and archaeal communities: 16S rRNA
  - Fungal communities: ITS (internal transcribed spacer between 18S and 5.8S rRNA genes)
- Sequenced using Illumina MiSeq or Ion Torrent
  - New: PacBio full-length sequencing provides better resolution
- Different from metagenomics, where the aim is to sequence all genes
  - Answers the questions who are there and what are they capable of doing

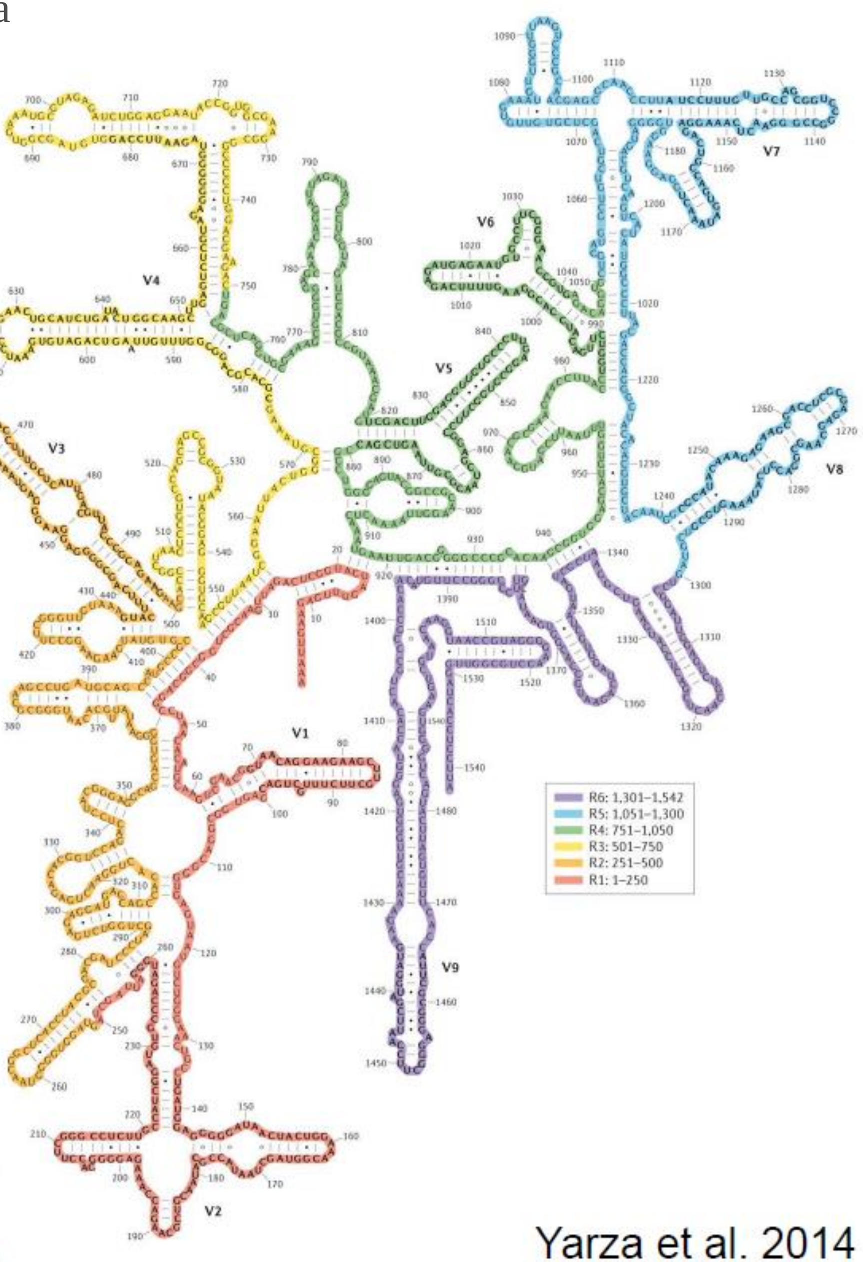
Slide by Dr Anu Mikkonen, University of Jyväskylä

Commonly used 16S rRNA gene amplicons are called by the variable regions they contain

- V1-V2
- (V3-)V4

Variable Regions of the 16S rRNA:





Yarza et al. 2014

Nature Reviews | Microbiology

UNIVERSITY OF JYVÄSKYLÄ

Main parts of microbial community analysis

- Preprocessing
  - Quality control, trim primers/adaptors and bad quality ends
  - Depending on data type:
    - MiSeq: Combine paired end reads to contigs
    - Ion Torrent: Combine samples and make a group file
  - Filter out bad quality sequences, remove identical sequences
  - Align sequences to reference template (e.g. SILVA)
  - Filter sequences based on alignment position, trim sequence alignment
  - Remove sequencing errors and chimeras
- Classification
  - Taxonomic assignment of sequences (e.g. SILVA for 16S, UNITE for ITS)
- Community analysis and visualization
  - Does community structure differ between sample groups?
  - Which taxa are differentially abundant between sample groups?

Main parts of ITS data analysis

- Preprocessing
  - Quality control, trim primers/adaptors and bad quality ends
  - Depending on data type:
    - MiSeq: Combine paired end reads to contigs
    - Ion Torrent: Combine samples and make a group file
  - Filter out bad quality sequences, remove identical sequences
  - ~~Align sequences to reference template (e.g. SILVA)~~
  - ~~Filter sequences based on alignment position, trim sequence alignment~~
  - Remove sequencing errors and chimeras
- Classification
  - Taxonomic assignment of sequences using the UNITE reference
- Community analysis and visualization
  - When running Generate input files for phyloseq set Type of data = ITS (AGC instead of OptiClust is used for clustering, because the sequence length varies widely)

Quality control of raw reads

Outline

- Different types of quality problems
- FASTQ file format
- Tools for checking read quality
- Tools for improving read quality

What and why?

- Potential problems
  - low confidence bases, Ns
  - adapters
  - …
- Knowing about potential problems in your data allows you to
  - correct for them before you spend a lot of time on analysis
  - take them into account when interpreting results

FASTQ file format

- Four lines per read:

```
@read name
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+ read name
!"*((((***+))%%%++)(%%%%).1***-+*"))**55CCF>>>>>>CCCCCCC65
```

- http://en.wikipedia.org/wiki/FASTQ_format
- Do **not** unzip FASTQ files, Chipster can cope with .gz files

Base qualities

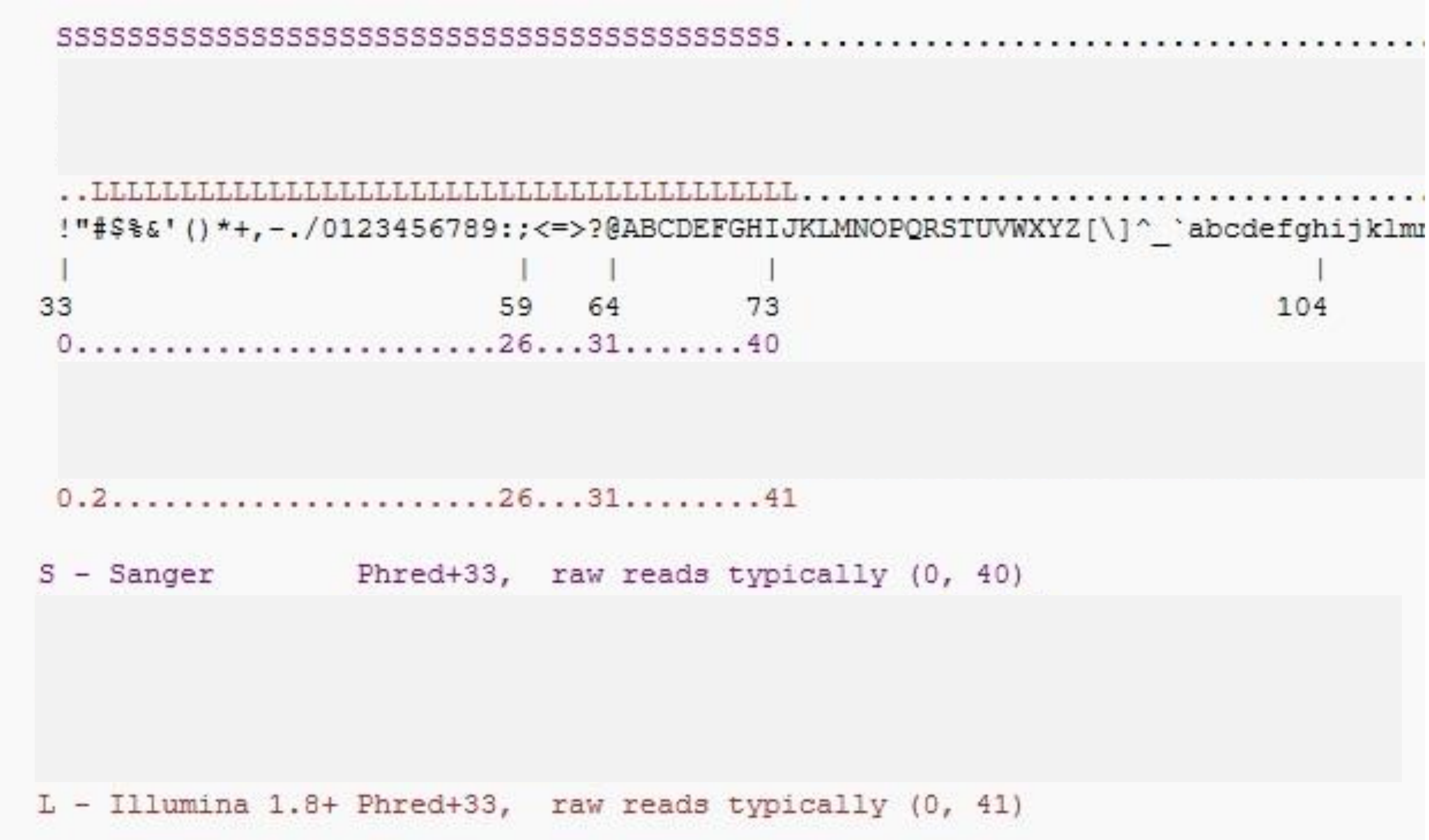
- If the quality of a base is 20, the probability that it is wrong is 0.01.
  - Phred quality score Q = -10 * log10 (probability that the base is wrong)

T C A G T A C T C G

40 40 40 40 40 40 40 40 37 35

- Sanger encoding: numbers are shown as ASCII characters so that 33 is added to the Phred score
  - E.g. 39 is encoded as H, the 72nd ASCII character (39+33 = 72)
  - Note that older Illumina data uses different encoding

Base quality encoding systems

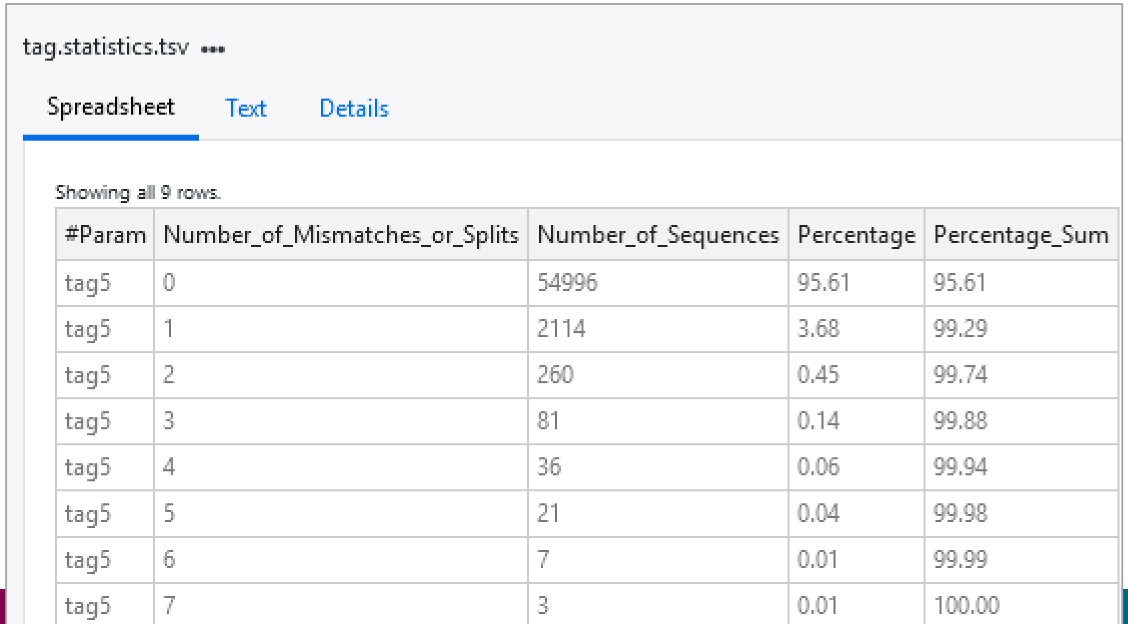


```
  SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS....................................

  ..LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL...................................
  !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklm
  |                         |    |        |                              |
 33                        59   64       73                            104
  0........................26...31.......40

  0.2......................26...31........41

 S - Sanger        Phred+33,  raw reads typically (0, 40)

 L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
```

http://en.wikipedia.org/wiki/FASTQ_format

Tools for checking sequence quality

- Read quality with MultiQC for many FASTQ files
  - runs FastQC for all the FASTQ files simultaneously
  - checks base quality and composition, duplication, Ns, k-mers, adapters,…
  - takes a tar package of all the FASTQ files as an input file
- Statistics for primers and adapters with TagCleaner
  - Given an adapter or primer sequence, checks how many reads have it (allowing mismatches)



Showing all 9 rows.

#Param	Number_of_Mismatches_or_Splits	Number_of_Sequences	Percentage	Percentage_Sum
tag5	0	54996	95.61	95.61
tag5	1	2114	3.68	99.29
tag5	2	260	0.45	99.74
tag5	3	81	0.14	99.88
tag5	4	36	0.06	99.94
tag5	5	21	0.04	99.98
tag5	6	7	0.01	99.99
tag5	7	3	0.01	100.00

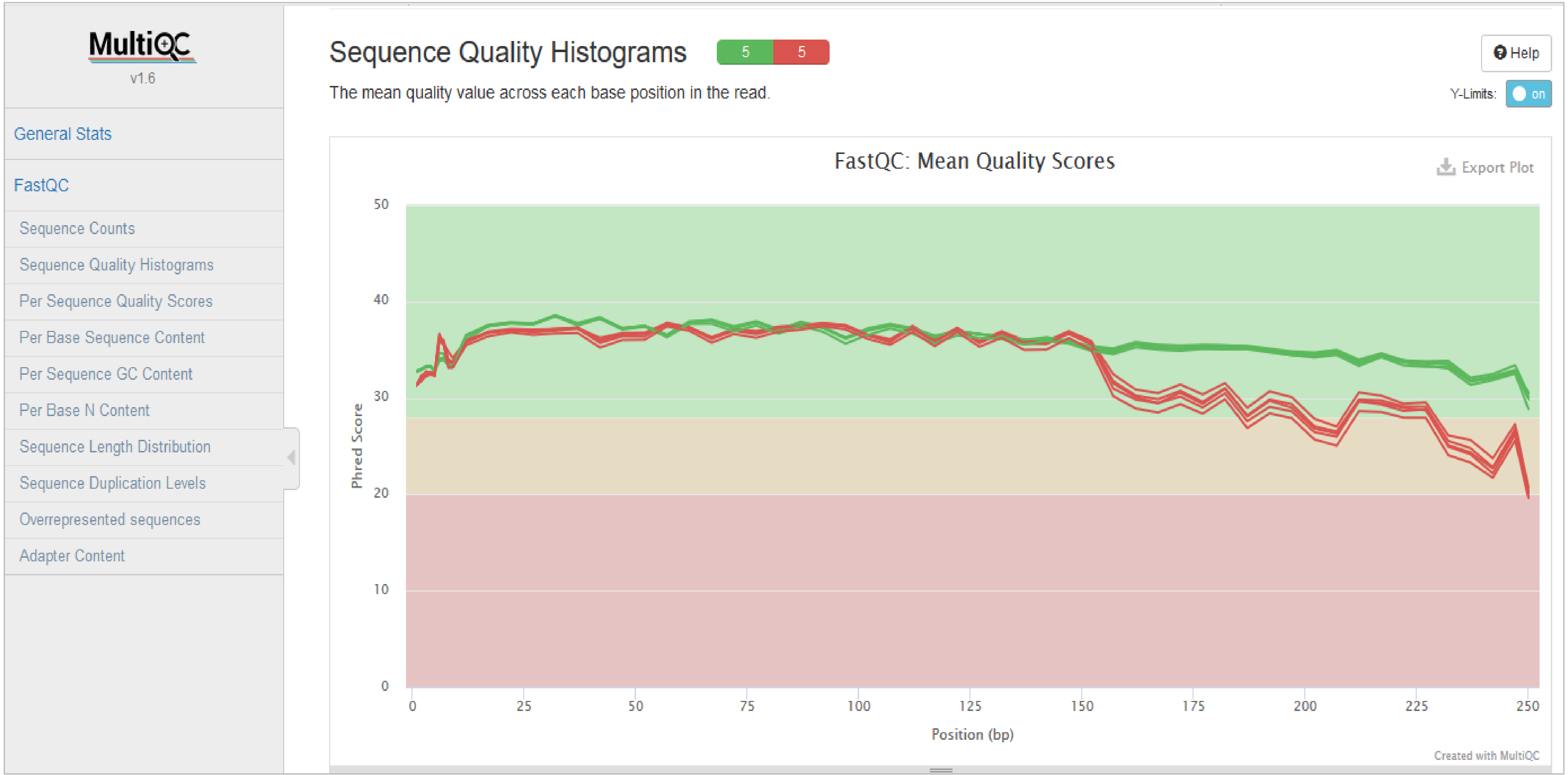
Making a Tar package of FASTQ files

- Use the tool Utilities / Make Tar package
- When your Tar package is ready, you can delete the original FASTQ files
  - If you want to look at the individual FASTQ files later, you can always open the Tar package using the tool Utilities / Extract .tar.gz file

MultiQC features

- Interactive plots
- Plots allow you to view the number or percentage of reads
- Traffic lights (they might not be suitable for your data!)
- Toolbox (click on the right side panel) allows you to
  - Highlight samples
  - Show only selected samples
  - Download plots
  - Rename samples
- Good tutorial video https://www.youtube.com/watch?v=qPbIlO_KWN0

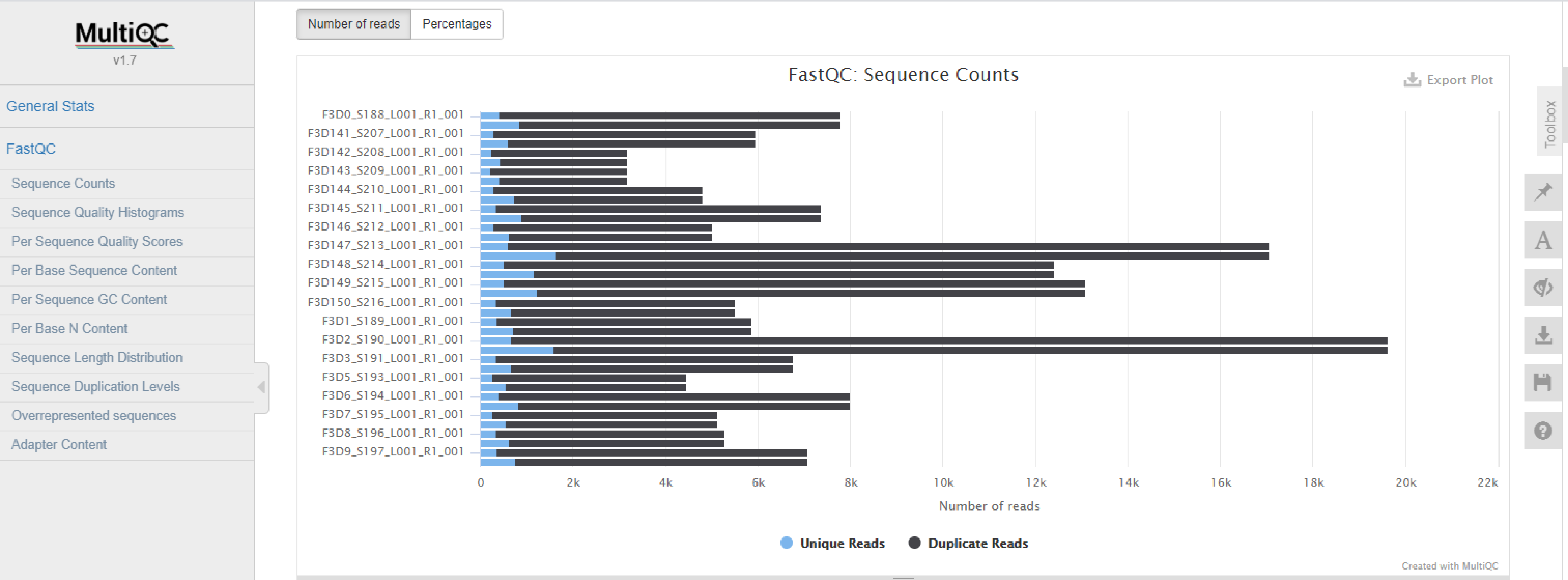
Per position base quality (MultiQC)



Sequence Quality Histograms 5 5

The mean quality value across each base position in the read.

Sequence counts (MultiQC)



What if there is a quality problem?

- You can either trim or filter reads
- Filtering removes the entire read, trimming removes only the bad quality bases
  - Note that trimming can remove the entire read, if all bases are bad
- Trimming makes reads shorter, which is not always optimal
- Paired end data: the matching order of the reads in the two files has to be preserved
  - If a read is removed, its pair has to be removed as well

Preprocessing tools for improving reads

- Trimmomatic and PRINSEQ
    - Can cope with paired end data
    - Trimmomatic is faster
- FastX
    - Does not take the pairing of reads into account
    - Can be used for trimming a given number of bases from either end of the reads
- TagCleaner
    - Removes primers and adapters allowing mismatches

Trimmomatic options in Chipster

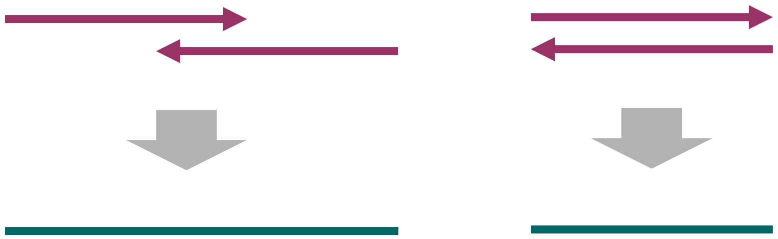
- Adapters
- Minimum quality
  - Per base, one base at a time or in a sliding window, from 3′ or 5′ end
  - Per base adaptive quality trimming (balance length and errors)
- Minimum mean read quality
- Trim x number of bases from beginning/ end
- Minimum read length after trimming
- Copes with paired end data

Combine paired reads to contigs

Outline

- How are reads joined by Mothur
- Things to take into account
- Result files

Reads are joined to contigs using Mothur's make.contigs tool



- Input file: Tar package of FASTQ files
- Creates a reverse complement of the reverse read
- Performs a Needleman alignment for the two reads
- What if the reads don't agree?
    - If one read has a base and the other has a gap, the quality of the base has to be at least 25 to be kept
    - If the bases differ, the quality difference has to be at least 6. If it is less, the base is set to N
- Problems if the read overlap is short or bad sequence quality
    - MiSeq 2x300 chemistry produces low quality ends
    - Sequence only short regions (~250 recommended by Patrick Schloss) so that you get full overlap
    - The USEARCH tool fastq_mergepairs followed by fastq_filter might work better

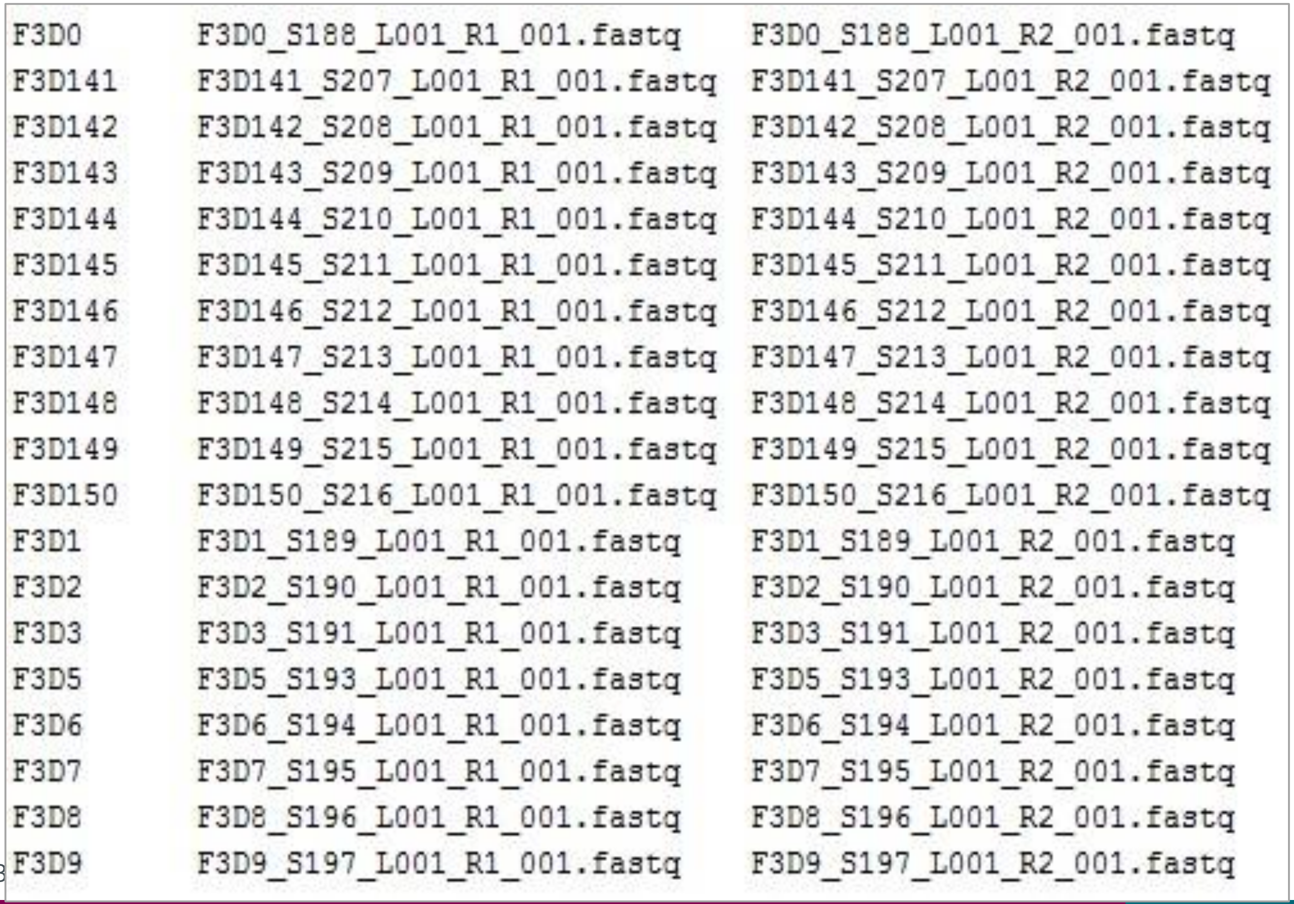
→VSEARCH alternatives will be tested

Result files of Mothur's make.contigs tool

- contigs.fasta.gz = contig sequences
- samples.fastqs.txt = FASTQ file assignment to samples
- contigs.groups = assignment of contigs to samples
- contig.numbers.txt = number of contig sequences in each sample
- contigs.summary.tsv = sequence information

samples.fastqs.txt

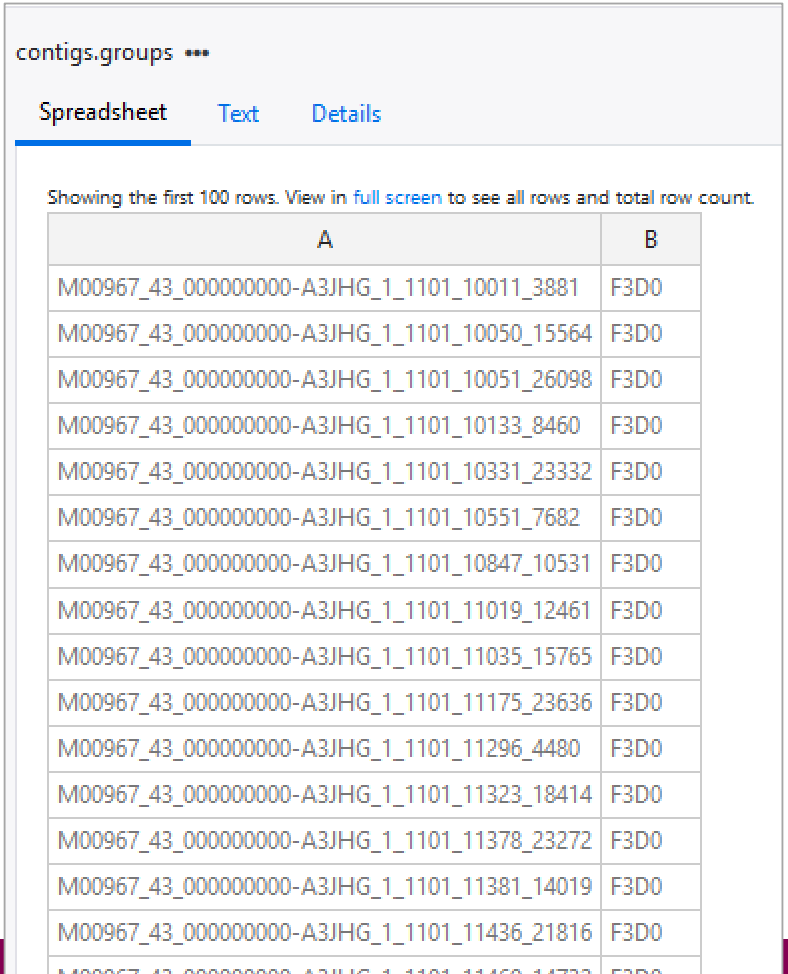
- Allows you to check if the FASTQ files were assinged correctly to each sample
- If the assignment is wrong, you can make this file yourself and give it as input



```
F3D0      F3D0_S188_L001_R1_001.fastq    F3D0_S188_L001_R2_001.fastq
F3D141    F3D141_S207_L001_R1_001.fastq  F3D141_S207_L001_R2_001.fastq
F3D142    F3D142_S208_L001_R1_001.fastq  F3D142_S208_L001_R2_001.fastq
F3D143    F3D143_S209_L001_R1_001.fastq  F3D143_S209_L001_R2_001.fastq
F3D144    F3D144_S210_L001_R1_001.fastq  F3D144_S210_L001_R2_001.fastq
F3D145    F3D145_S211_L001_R1_001.fastq  F3D145_S211_L001_R2_001.fastq
F3D146    F3D146_S212_L001_R1_001.fastq  F3D146_S212_L001_R2_001.fastq
F3D147    F3D147_S213_L001_R1_001.fastq  F3D147_S213_L001_R2_001.fastq
F3D148    F3D148_S214_L001_R1_001.fastq  F3D148_S214_L001_R2_001.fastq
F3D149    F3D149_S215_L001_R1_001.fastq  F3D149_S215_L001_R2_001.fastq
F3D150    F3D150_S216_L001_R1_001.fastq  F3D150_S216_L001_R2_001.fastq
F3D1      F3D1_S189_L001_R1_001.fastq    F3D1_S189_L001_R2_001.fastq
F3D2      F3D2_S190_L001_R1_001.fastq    F3D2_S190_L001_R2_001.fastq
F3D3      F3D3_S191_L001_R1_001.fastq    F3D3_S191_L001_R2_001.fastq
F3D5      F3D5_S193_L001_R1_001.fastq    F3D5_S193_L001_R2_001.fastq
F3D6      F3D6_S194_L001_R1_001.fastq    F3D6_S194_L001_R2_001.fastq
F3D7      F3D7_S195_L001_R1_001.fastq    F3D7_S195_L001_R2_001.fastq
F3D8      F3D8_S196_L001_R1_001.fastq    F3D8_S196_L001_R2_001.fastq
F3D9      F3D9_S197_L001_R1_001.fastq    F3D9_S197_L001_R2_001.fastq
```

contigs.groups

- All our sequences are now in one FASTA file. The groups file tells which sequence comes from which sample.



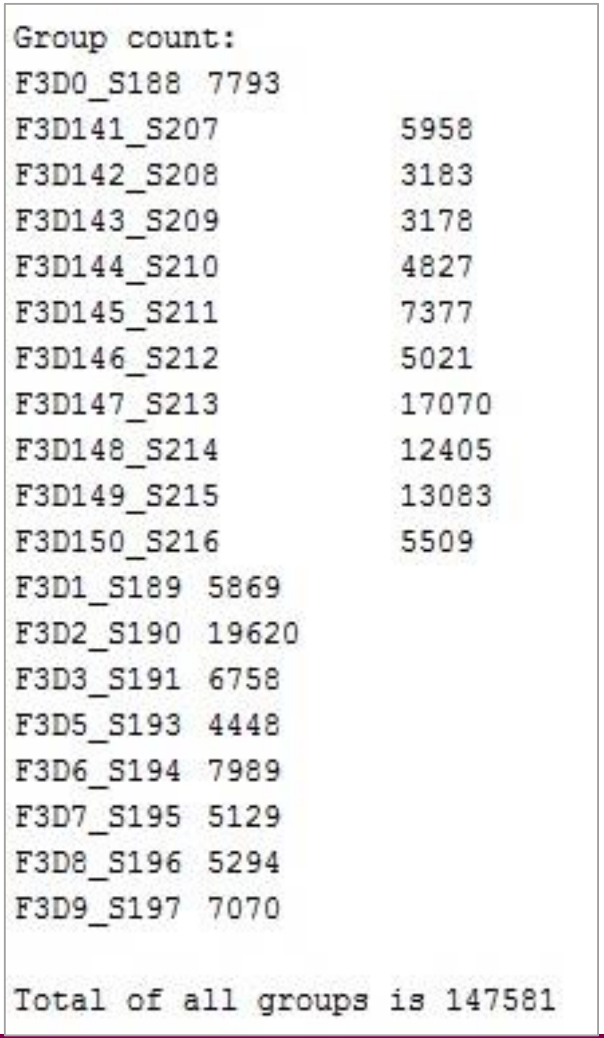
contigs.groups

Showing the first 100 rows. View in full screen to see all rows and total row count.

A	B
M00967_43_000000000-A3JHG_1_1101_10011_3881	F3D0
M00967_43_000000000-A3JHG_1_1101_10050_15564	F3D0
M00967_43_000000000-A3JHG_1_1101_10051_26098	F3D0
M00967_43_000000000-A3JHG_1_1101_10133_8460	F3D0
M00967_43_000000000-A3JHG_1_1101_10331_23332	F3D0
M00967_43_000000000-A3JHG_1_1101_10551_7682	F3D0
M00967_43_000000000-A3JHG_1_1101_10847_10531	F3D0
M00967_43_000000000-A3JHG_1_1101_11019_12461	F3D0
M00967_43_000000000-A3JHG_1_1101_11035_15765	F3D0
M00967_43_000000000-A3JHG_1_1101_11175_23636	F3D0
M00967_43_000000000-A3JHG_1_1101_11296_4480	F3D0
M00967_43_000000000-A3JHG_1_1101_11323_18414	F3D0
M00967_43_000000000-A3JHG_1_1101_11378_23272	F3D0
M00967_43_000000000-A3JHG_1_1101_11381_14019	F3D0
M00967_43_000000000-A3JHG_1_1101_11436_21816	F3D0

contig.numbers.txt

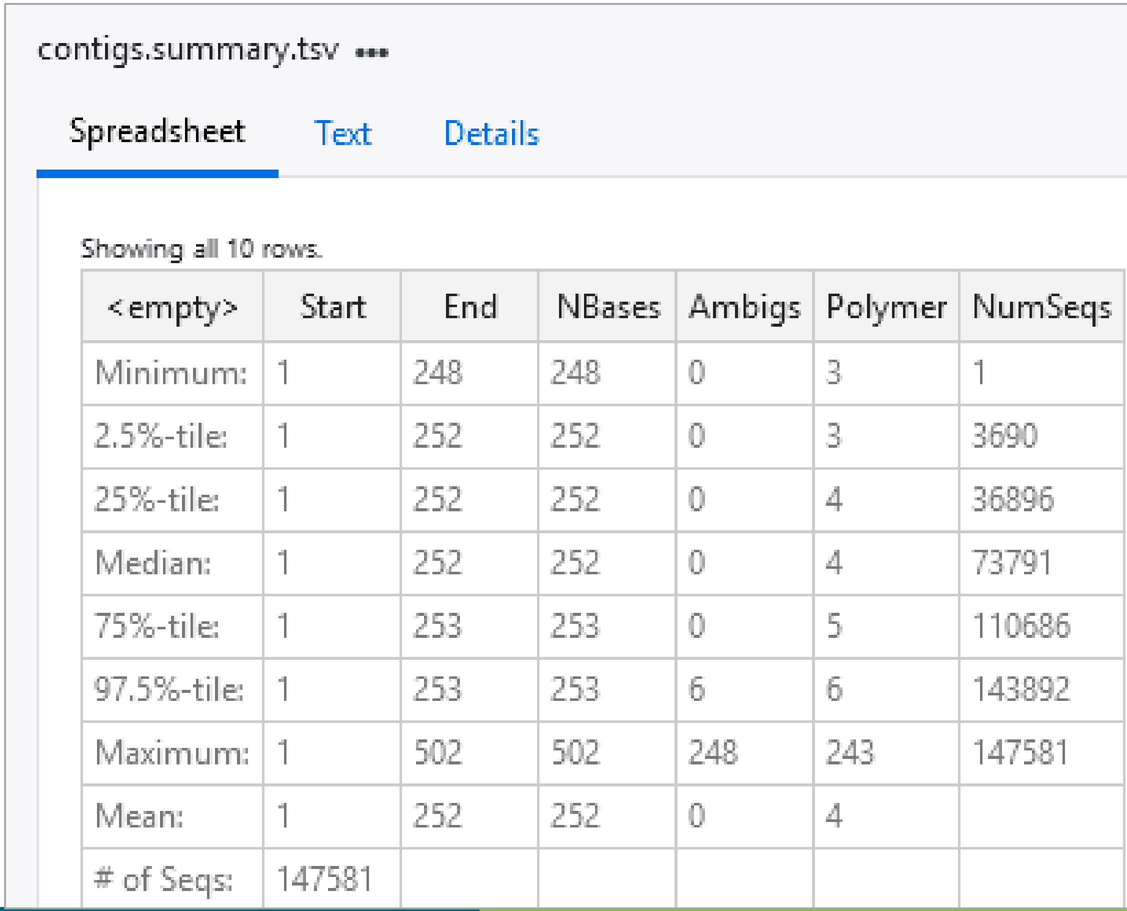
- Number of contig sequences per sample and in total



```
Group count:
F3D0_S188  7793
F3D141_S207          5958
F3D142_S208          3183
F3D143_S209          3178
F3D144_S210          4827
F3D145_S211          7377
F3D146_S212          5021
F3D147_S213          17070
F3D148_S214          12405
F3D149_S215          13083
F3D150_S216          5509
F3D1_S189  5869
F3D2_S190  19620
F3D3_S191  6758
F3D5_S193  4448
F3D6_S194  7989
F3D7_S195  5129
F3D8_S196  5294
F3D9_S197  7070

Total of all groups is 147581
```

contigs.summary.tsv

- Number of sequences: total and unique
- Stats (min, max, mean, median and quantiles) of
  - number of bases
  - number of ambiguous bases
  - start and end positions
  - homopolymer length



Showing all 10 rows.

<empty>	Start	End	NBases	Ambigs	Polymer	NumSeqs
Minimum:	1	248	248	0	3	1
2.5%-tile:	1	252	252	0	3	3690
25%-tile:	1	252	252	0	4	36896
Median:	1	252	252	0	4	73791
75%-tile:	1	253	253	0	5	110686
97.5%-tile:	1	253	253	6	6	143892
Maximum:	1	502	502	248	243	147581
Mean:	1	252	252	0	4	
# of Seqs:	147581					

How to start with Ion Torrent data?

- Create a Tar package
- Perform quality control with MultiQC (and TagCleaner if needed)
- Trim reads with FastX to a suitable length (use the Tar package as input)
- (Single end data, so no need to combine paired reads to contigs)
- Use the tool Combine FASTQ or FASTA files and make a group file to
  - convert FASTQ to FASTA
  - merge all the samples in one file
  - create the Mothur groups file
- Continue like with MiSeq data

Filter contigs and remove identical sequences

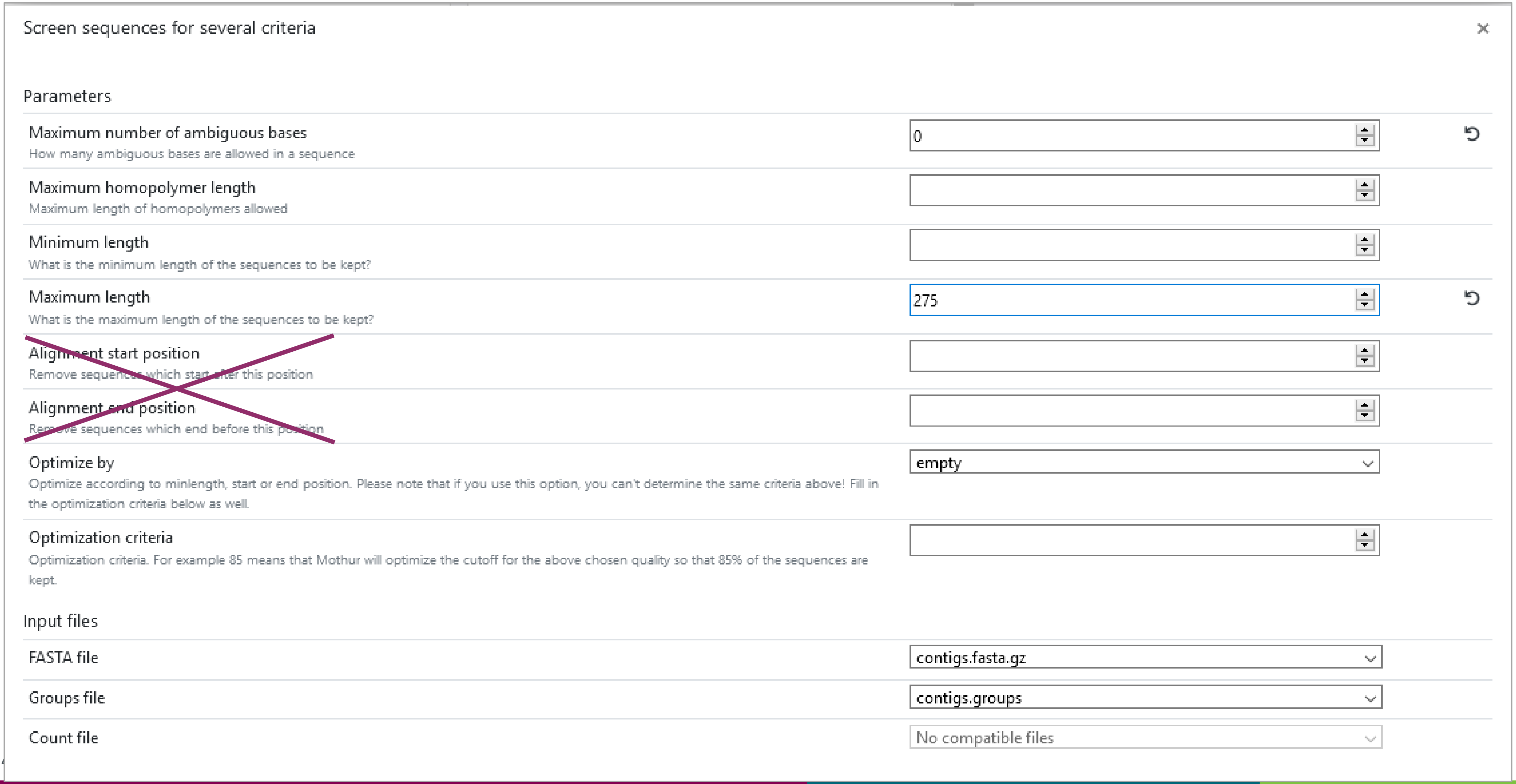
Outline

- How to filter contigs based on length etc
- Why identical sequences need to be removed
- Mothur count file format

Filter contigs based on length, ambiguous bases and homopolymers

- Tool Screen sequences for several criteria, based on Mothur screens.seqs command
  - the same tool is used after reference alignment to filter based on alignment start and end position
- Two options for screening based on length, start and end
  - set the minimum and maximum values manually
  - select optimize and tell what percentage of sequences you want to keep
- Give contigs fasta file and groups file as input
- Set the parameters according to the stats in the contigs.summary.tsv
- Result files:
  - screened.fasta.gz = screened sequences
  - screened.groups = sample assignment of the screened sequences
  - summary.screened.tsv = sequence information

Screen sequences for several criteria



Screen sequences for several criteria

Parameters

Maximum number of ambiguous bases How many ambiguous bases are allowed in a sequence	0
Maximum homopolymer length Maximum length of homopolymers allowed	
Minimum length What is the minimum length of the sequences to be kept?	
Maximum length What is the maximum length of the sequences to be kept?	275
~~Alignment start position~~ ~~Remove sequences which start after this position~~	
~~Alignment end position~~ ~~Remove sequences which end before this position~~	
Optimize by Optimize according to minlength, start or end position. Please note that if you use this option, you can't determine the same criteria above! Fill in the optimization criteria below as well.	empty
Optimization criteria Optimization criteria. For example 85 means that Mothur will optimize the cutoff for the above chosen quality so that 85% of the sequences are kept.	

Input files

FASTA file	contigs.fasta.gz
Groups file	contigs.groups
Count file	No compatible files

Remove identical sequences

- The fasta file contains many identical sequences
- Aligning the same sequence to the reference would be computationally wasteful

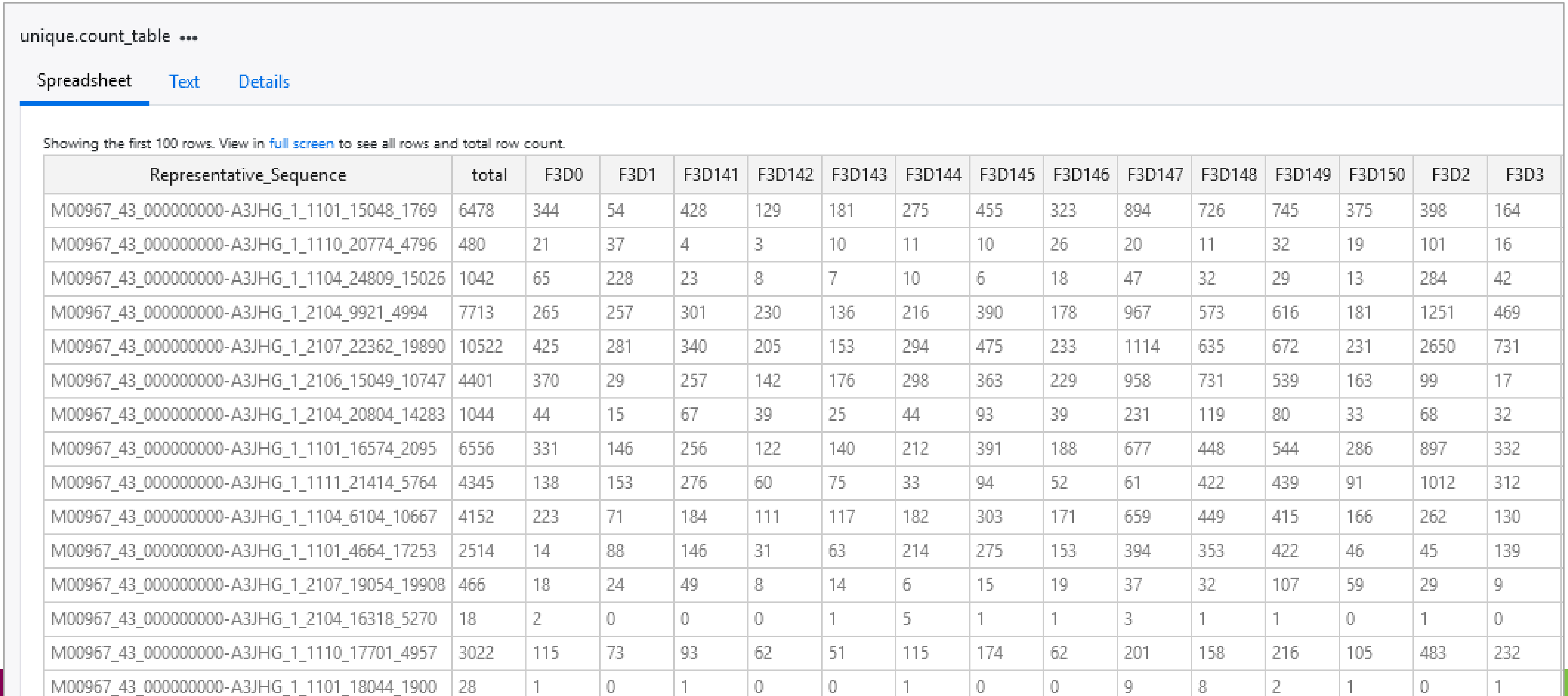
→ We remove identical sequences and keep only one representative in the fasta file

  - keep track of how many sequences the representative represents, in the different samples

- Tool Extract unique sequences, based on Mothur unique.seqs and count.seqs commands
- Give fasta file and groups file as input
- Output files
  - unique.fasta = unique sequences
  - unique.count_table = how many represented sequences are in each sample
  - unique.summary.tsv = sequence information

unique.count_table

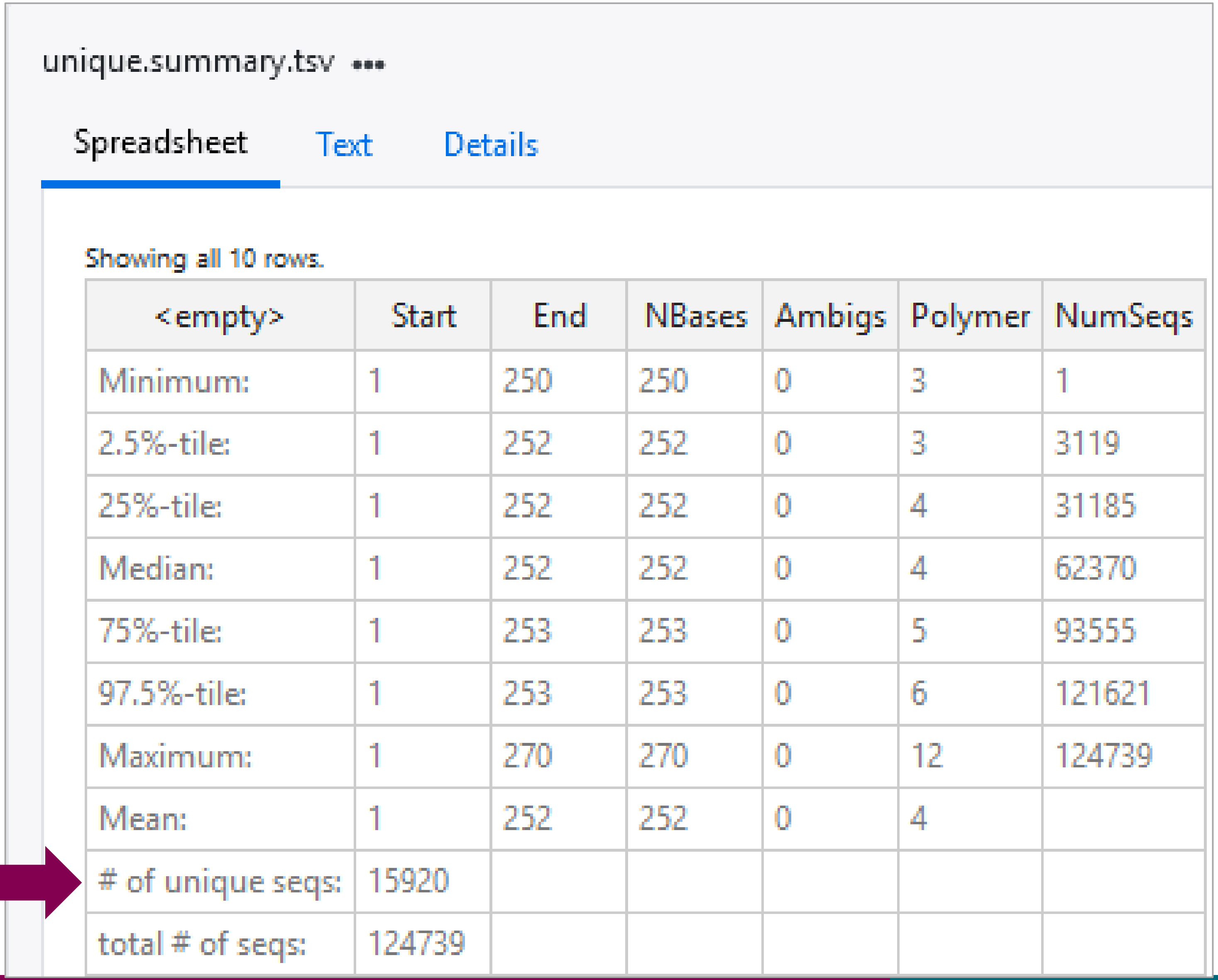
- Rows = names of unique representative sequences
- Columns = samples
- Cells = how many times the representative sequence occurs in each sample



Showing the first 100 rows. View in full screen to see all rows and total row count.

Representative_Sequence	total	F3D0	F3D1	F3D141	F3D142	F3D143	F3D144	F3D145	F3D146	F3D147	F3D148	F3D149	F3D150	F3D2	F3D3
M00967_43_000000000-A3JHG_1_1101_15048_1769	6478	344	54	428	129	181	275	455	323	894	726	745	375	398	164
M00967_43_000000000-A3JHG_1_1110_20774_4796	480	21	37	4	3	10	11	10	26	20	11	32	19	101	16
M00967_43_000000000-A3JHG_1_1104_24809_15026	1042	65	228	23	8	7	10	6	18	47	32	29	13	284	42
M00967_43_000000000-A3JHG_1_2104_9921_4994	7713	265	257	301	230	136	216	390	178	967	573	616	181	1251	469
M00967_43_000000000-A3JHG_1_2107_22362_19890	10522	425	281	340	205	153	294	475	233	1114	635	672	231	2650	731
M00967_43_000000000-A3JHG_1_2106_15049_10747	4401	370	29	257	142	176	298	363	229	958	731	539	163	99	17
M00967_43_000000000-A3JHG_1_2104_20804_14283	1044	44	15	67	39	25	44	93	39	231	119	80	33	68	32
M00967_43_000000000-A3JHG_1_1101_16574_2095	6556	331	146	256	122	140	212	391	188	677	448	544	286	897	332
M00967_43_000000000-A3JHG_1_1111_21414_5764	4345	138	153	276	60	75	33	94	52	61	422	439	91	1012	312
M00967_43_000000000-A3JHG_1_1104_6104_10667	4152	223	71	184	111	117	182	303	171	659	449	415	166	262	130
M00967_43_000000000-A3JHG_1_1101_4664_17253	2514	14	88	146	31	63	214	275	153	394	353	422	46	45	139
M00967_43_000000000-A3JHG_1_2107_19054_19908	466	18	24	49	8	14	6	15	19	37	32	107	59	29	9
M00967_43_000000000-A3JHG_1_2104_16318_5270	18	2	0	0	0	1	5	1	1	3	1	1	0	1	0
M00967_43_000000000-A3JHG_1_1110_17701_4957	3022	115	73	93	62	51	115	174	62	201	158	216	105	483	232
M00967_43_000000000-A3JHG_1_1101_18044_1900	28	1	0	1	0	0	1	0	0	9	8	2	1	0	1

unique.summary.tsv



Showing all 10 rows.

<empty>	Start	End	NBases	Ambigs	Polymer	NumSeqs
Minimum:	1	250	250	0	3	1
2.5%-tile:	1	252	252	0	3	3119
25%-tile:	1	252	252	0	4	31185
Median:	1	252	252	0	4	62370
75%-tile:	1	253	253	0	5	93555
97.5%-tile:	1	253	253	0	6	121621
Maximum:	1	270	270	0	12	124739
Mean:	1	252	252	0	4	
# of unique seqs:	15920					
total # of seqs:	124739					

Align sequences to reference template alignment

Outline

- SILVA reference template alignment
- Alignment steps
- How to improve and speed up the alignment
- Alignment file format

SILVA reference template alignment

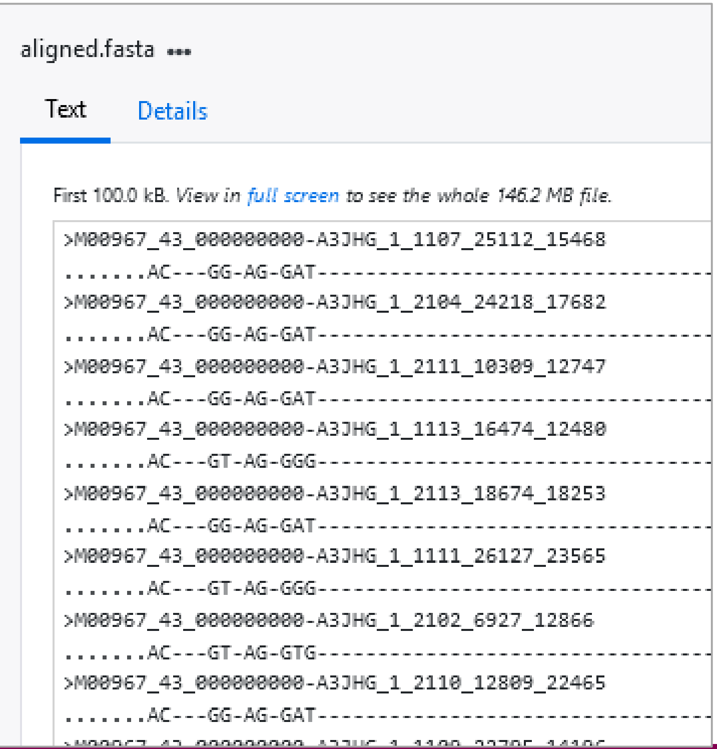
- In order to identify the sequences we align them to a reference template alignment
- Chipster uses the full SILVA template, but you can also give your own
- The current SILVA version is 138.1
  - Contains 146 601 sequences: 128 884 bacteria, 2846 archaea, and 14 871 eukarya
  - the alignment is 50 000 columns long so that it is compatible with 18S rRNA sequences and archaeal 16S rRNA sequences
  - in order to make alignment process faster, you can indicate which region of the SILVA template alignment matches the area you amplified
  - In order to get the SILVA coordinates of that area, you can align a small number of samples first
- https://mothur.org/wiki/Silva_reference_files

Aligning sequences to template alignment

- Tool Align sequences to reference, based on Mothur align.seqs and pcr.seqs commands
- Give unique.fasta.gz and unique.count_table as input
- Three steps
  - find the closest template sequence for the query sequence using K-mer search with 8mers
  - align the query and the de-gapped template sequence using Needleman-Wunsch pairwise alignment
  - re-insert gaps to the query and template pairwise alignment using the NAST algorithm so that the query sequence alignment is compatible with the original template alignment
- Speed depends on the number and length of the query and template sequences
- Limit the alignment to the template region which corresponds to the part of the 16S rRNA gene you amplified → better alignment quality, less space needed

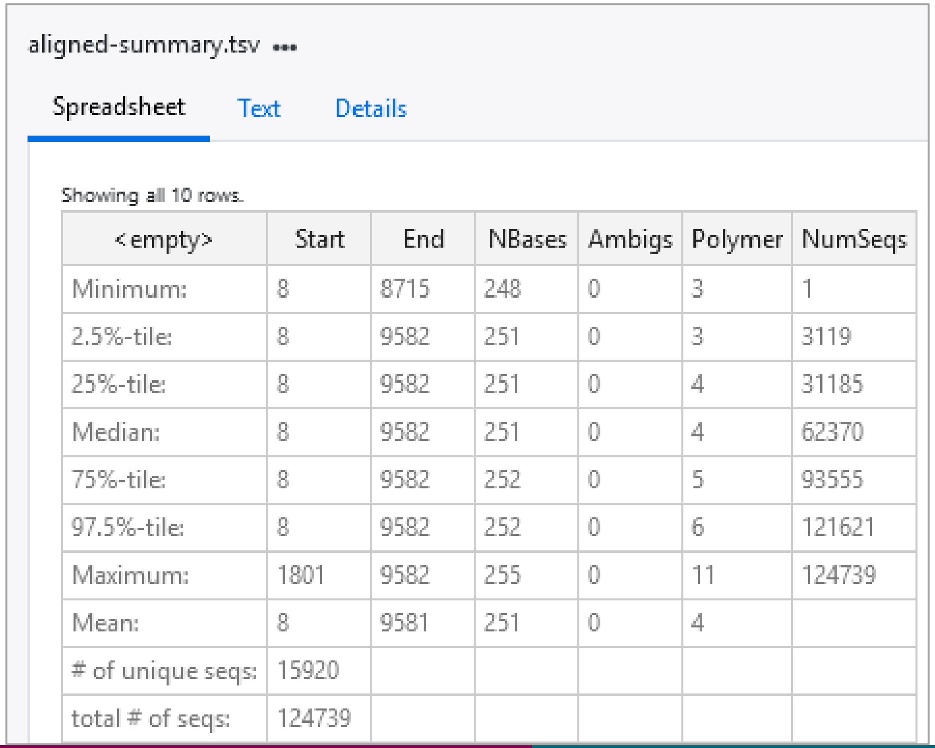
Result files

- aligned.fasta.gz = aligned sequences
  - periods lead to the first base in the sequence and follow the last base of the sequence
- custom.reference.summary.tsv = information on the region of the reference used
- aligned-summary.tsv = aligned sequence information



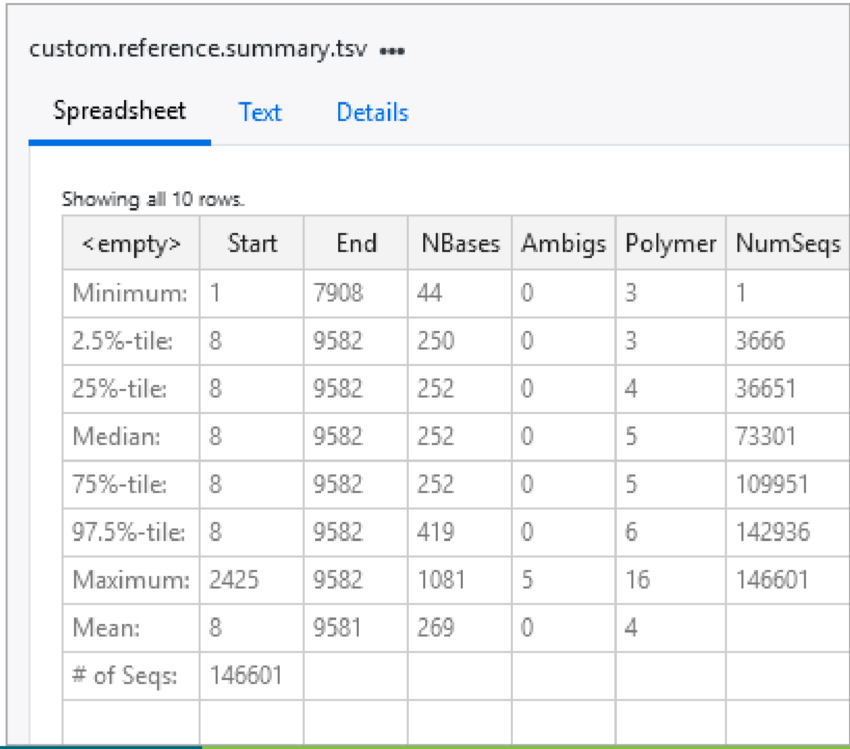
First 100.0 kB. *View in full screen to see the whole 146.2 MB file.*

```
>M00967_43_000000000-A3JHG_1_1107_25112_15468
.......AC---GG-AG-GAT--------------------------------
>M00967_43_000000000-A3JHG_1_2104_24218_17682
.......AC---GG-AG-GAT--------------------------------
>M00967_43_000000000-A3JHG_1_2111_10309_12747
.......AC---GG-AG-GAT--------------------------------
>M00967_43_000000000-A3JHG_1_1113_16474_12480
.......AC---GT-AG-GGG--------------------------------
>M00967_43_000000000-A3JHG_1_2113_18674_18253
.......AC---GG-AG-GAT--------------------------------
>M00967_43_000000000-A3JHG_1_1111_26127_23565
.......AC---GT-AG-GGG--------------------------------
>M00967_43_000000000-A3JHG_1_2102_6927_12866
.......AC---GT-AG-GTG--------------------------------
>M00967_43_000000000-A3JHG_1_2110_12809_22465
.......AC---GG-AG-GAT--------------------------------
```



Showing all 10 rows.

<empty>	Start	End	NBases	Ambigs	Polymer	NumSeqs
Minimum:	8	8715	248	0	3	1
2.5%-tile:	8	9582	251	0	3	3119
25%-tile:	8	9582	251	0	4	31185
Median:	8	9582	251	0	4	62370
75%-tile:	8	9582	252	0	5	93555
97.5%-tile:	8	9582	252	0	6	121621
Maximum:	1801	9582	255	0	11	124739
Mean:	8	9581	251	0	4	
# of unique seqs:	15920					
total # of seqs:	124739					



Showing all 10 rows.

<empty>	Start	End	NBases	Ambigs	Polymer	NumSeqs
Minimum:	1	7908	44	0	3	1
2.5%-tile:	8	9582	250	0	3	3666
25%-tile:	8	9582	252	0	4	36651
Median:	8	9582	252	0	5	73301
75%-tile:	8	9582	252	0	5	109951
97.5%-tile:	8	9582	419	0	6	142936
Maximum:	2425	9582	1081	5	16	146601
Mean:	8	9581	269	0	4	
# of Seqs:	146601					

Filter and trim aligned sequences

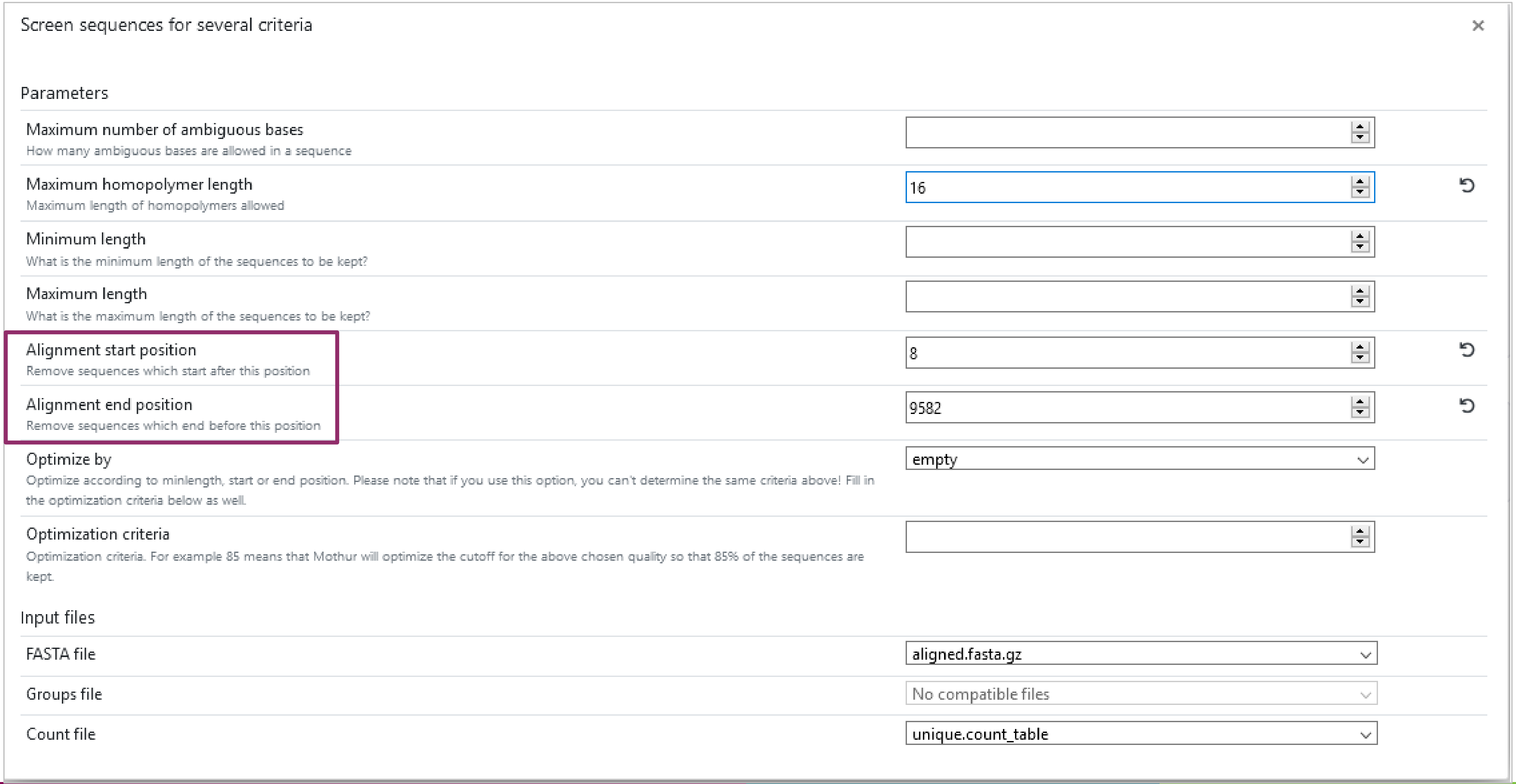
Outline

- Filter sequences based on alignment start and end position
- Trim sequence alignment
- Remove identical sequences

Filter aligned sequences

- All the aligned sequences should overlap the same alignment coordinates
- Remove deviants by filtering based on the alignment start and end position
  - Check aligned-summary.tsv
- Remove also sequences which have homopolymers longer than those in the reference
  - Check custom.reference.summary.tsv
- Tool Screen sequences for several criteria, based on Mothur screens.seqs command
- Give aligned.fasta.gz and unique.count_table as input files
- Result files
  - screened.fasta.gz = screened sequences
  - screened.count_table = updated count_table
  - summary.screened.tsv = sequence information

Parameters for filtering aligned sequences



Screen sequences for several criteria

Parameters

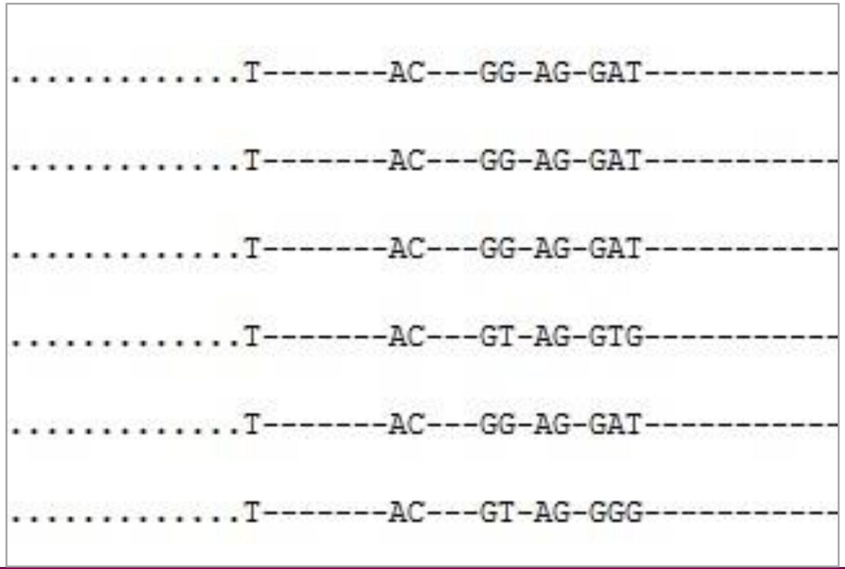
Maximum number of ambiguous bases How many ambiguous bases are allowed in a sequence	
Maximum homopolymer length Maximum length of homopolymers allowed	16
Minimum length What is the minimum length of the sequences to be kept?	
Maximum length What is the maximum length of the sequences to be kept?	
Alignment start position Remove sequences which start after this position	8
Alignment end position Remove sequences which end before this position	9582
Optimize by Optimize according to minlength, start or end position. Please note that if you use this option, you can't determine the same criteria above! Fill in the optimization criteria below as well.	empty
Optimization criteria Optimization criteria. For example 85 means that Mothur will optimize the cutoff for the above chosen quality so that 85% of the sequences are kept.	

Input files

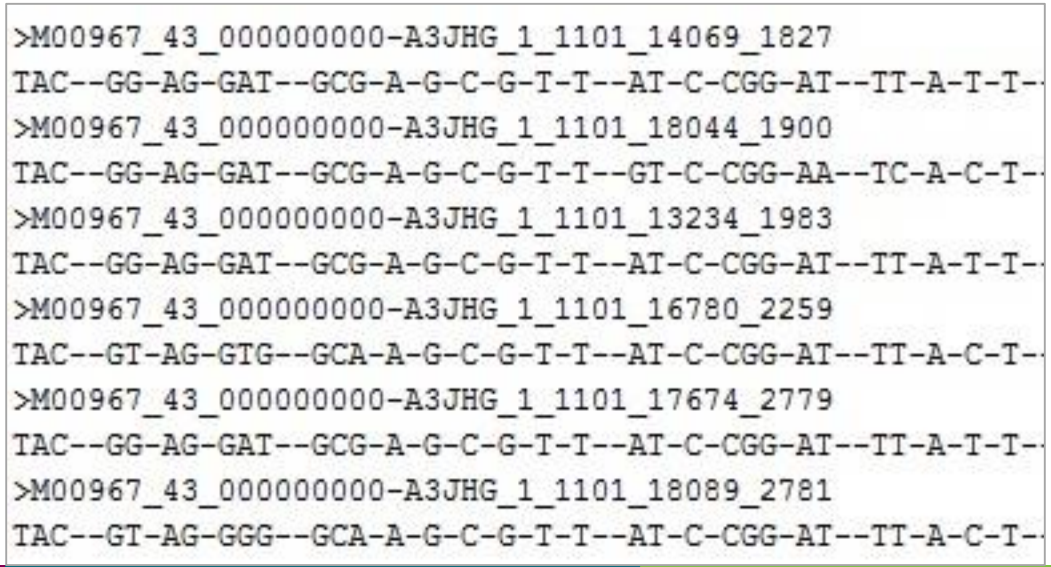
FASTA file	aligned.fasta.gz
Groups file	No compatible files
Count file	unique.count_table

Trim sequence alignment for overhangs and empty columns

- We remove overhangs (columns containing .) and keep the common alignment region
- Gap columns (where all the characters are –) have no information, so we remove them
  - makes distance calculation faster
- Removing alignment columns can create identical sequences → need to remove them
- Tool Filter sequence alignment, based on Mothur filter.seqs and unique.seqs commands
- Give screened.fasta.gz and screened.count_table as input files



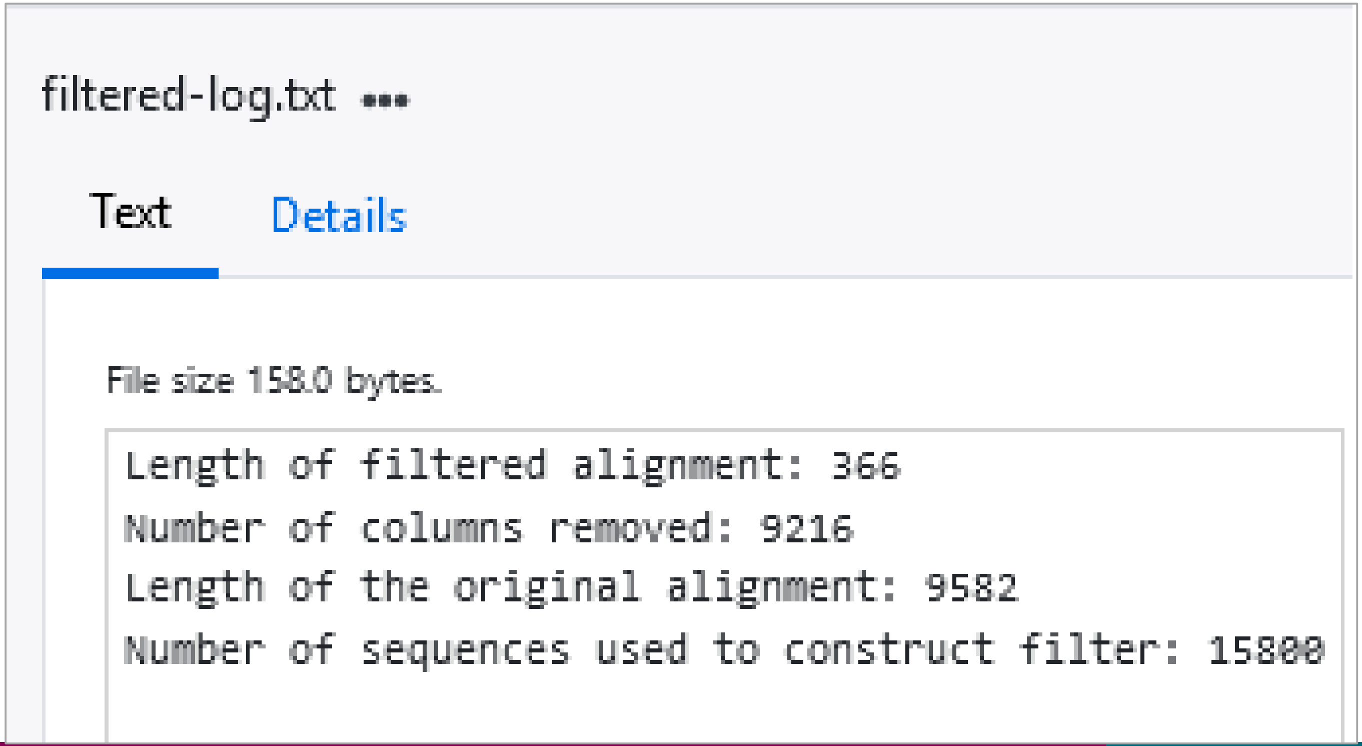
```
.............T-------AC---GG-AG-GAT-----------

.............T-------AC---GG-AG-GAT-----------

.............T-------AC---GG-AG-GAT-----------

.............T-------AC---GT-AG-GTG-----------

.............T-------AC---GG-AG-GAT-----------

.............T-------AC---GT-AG-GGG-----------
```



```
>M00967_43_000000000-A3JHG_1_1101_14069_1827
TAC--GG-AG-GAT--GCG-A-G-C-G-T-T--AT-C-CGG-AT--TT-A-T-T-
>M00967_43_000000000-A3JHG_1_1101_18044_1900
TAC--GG-AG-GAT--GCG-A-G-C-G-T-T--GT-C-CGG-AA--TC-A-C-T-
>M00967_43_000000000-A3JHG_1_1101_13234_1983
TAC--GG-AG-GAT--GCG-A-G-C-G-T-T--AT-C-CGG-AT--TT-A-T-T-
>M00967_43_000000000-A3JHG_1_1101_16780_2259
TAC--GT-AG-GTG--GCA-A-G-C-G-T-T--AT-C-CGG-AT--TT-A-C-T-
>M00967_43_000000000-A3JHG_1_1101_17674_2779
TAC--GG-AG-GAT--GCG-A-G-C-G-T-T--AT-C-CGG-AT--TT-A-T-T-
>M00967_43_000000000-A3JHG_1_1101_18089_2781
TAC--GT-AG-GGG--GCA-A-G-C-G-T-T--AT-C-CGG-AT--TT-A-C-T-
```

Result files

- filtered-unique.fasta.gz = trimmed aligned sequences
- filtered-unique.count_table = updated count_table
- filtered-unique-summary.tsv = sequence information
- filtered-log.txt = how many alignment columns were removed



File size 158.0 bytes.

```
Length of filtered alignment: 366
Number of columns removed: 9216
Length of the original alignment: 9582
Number of sequences used to construct filter: 15800
```

Remove sequencing errors and chimeras

Outline

- How preclustering works
- What are chimeras and how to remove them?

Precluster very similar sequences

- Assumes that abundant sequences are more likely to generate sequencing errors
  - ranks sequences in order of their abundance
  - walks through the list looking for rarer sequences which differ only by x number of bases from the original sequence (allow 1 mismatch for every 100 bp of sequence)
  - merges those that are within the threshold
- Tool Precluster aligned sequences, based on Mothur precluster.seqs command
- Give filtered-unique.fasta.gz and filtered-unique.count_table as input files
- Result files
  - preclustered.fasta.gz = preclustered aligned sequences
  - preclustered.count_table = updated count_table
  - preclustered-summary.tsv = sequence information

Remove chimeras

- Chimera = artifact sequence formed by two biological sequences
  - incomplete extension during PCR allows subsequent PCR cycles to use a partially extended strand to bind to the template of a similar sequence.
  - the partially extended strand then acts as a primer to extend and form a chimeric sequence.
  - as many as 30% of the sequences from mixed template environmental samples may be chimeric.
- Tool Remove chimeric sequences, based on Mothur chimera.uchime and chimera.vsearch
- You can either use a reference or detect chimeras *de novo*
  - Reference is the bacterial subset of the Silva Gold 16S rRNA
  - *De novo* approach uses the more abundant sequences in your data as the reference
- Dereplicate = should we remove a chimera only from the sample where it was spotted?
  - True = only from that sample ("do not replicate")
  - False = from all samples ("replicate to other samples")
- Give preclustered.fasta.gz and preclustered.count_table file as input files

Chimera removal results

- Result files
  - chimeras.removed.fasta.gz = aligned sequences
  - chimeras.removed.count_table = updated count_table
  - chimeras.removed.summary.tsv = sequence information
- Results depend heavily on the method and reference used. Example:
  - 6022 unique sequences to start with
  - 5283 after chimera removal with VSEARCH and SILVA gold (29 s)
  - 2467 after chimera removal with VSEARCH and *de novo* (4 s)
  - 5323 after chimera removal with UCHIME and SILVA gold (23 min)
  - 5023 after chimera removal with UCHIME and *de novo* (19 s)

Classify sequences to taxonomic units

Outline

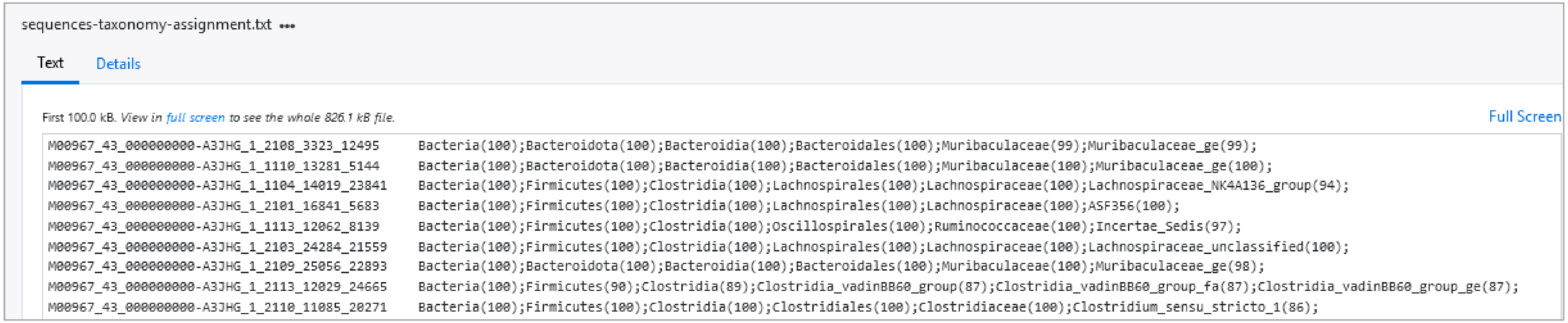
- Tools for assigning sequences to taxonomies
- Wang method
- File formats
    - Taxonomy assignment file
    - Classification summary file

Assign sequences to taxonomy

- Tools Classify 16S or 18S sequences to taxonomic units using Silva and Classify ITS sequences to taxonomic units using UNITE, based on Mothur classify.seqs command
  - you can also use your own taxonomy by providing reference fasta and taxonomy outline file.
- Wang method
  - looks at the query sequence k-mer by k-mer
  - calculates the probability a sequence from a given taxonomy would contain a specific k-mer
  - calculates the probability a query sequence would be in a given taxonomy based on the k-mers it contains, and assigns the query sequence to the taxonomy with the highest probability
  - calculates bootstrap confidence score for the assignment (chooses randomly 1/8 of the k-mers in the query), by default 100 iterations
  - if the confidence is <80%, assignment will revert to higher level
- Give chimeras.removed.fasta.gz and chimeras.removed.count_table as input files

Classification result files

- sequences-taxonomy-assignment.txt = sequence name and taxonomy

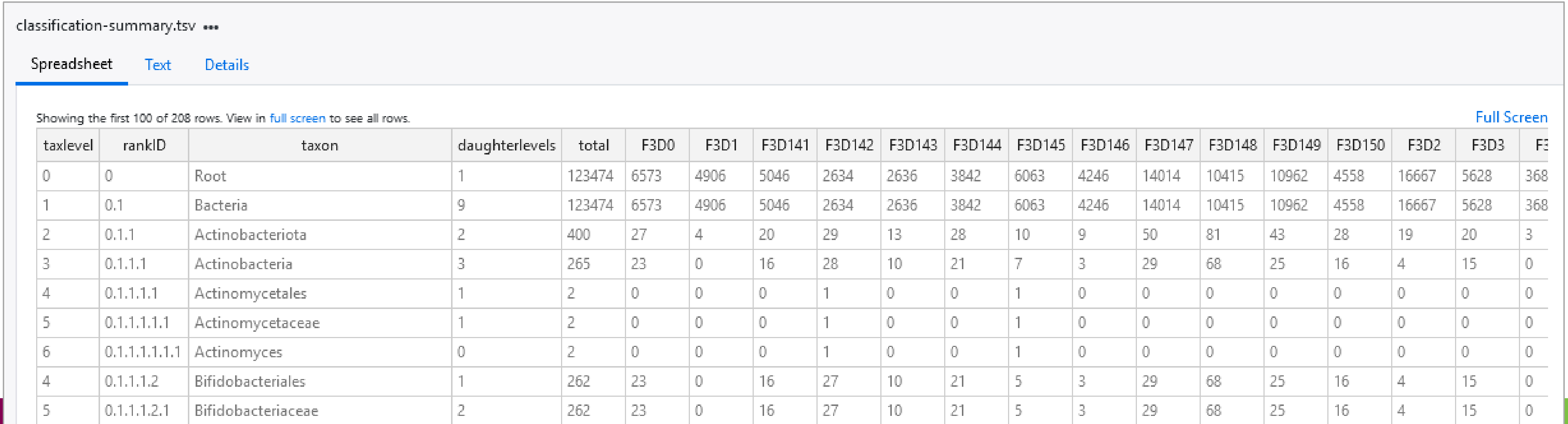


sequences-taxonomy-assignment.txt •••

First 100.0 kB. *View in full screen to see the whole 826.1 kB file.*

```
M00967_43_000000000-A3JHG_1_2108_3323_12495	Bacteria(100);Bacteroidota(100);Bacteroidia(100);Bacteroidales(100);Muribaculaceae(99);Muribaculaceae_ge(99);
M00967_43_000000000-A3JHG_1_1110_13281_5144	Bacteria(100);Bacteroidota(100);Bacteroidia(100);Bacteroidales(100);Muribaculaceae(100);Muribaculaceae_ge(100);
M00967_43_000000000-A3JHG_1_1104_14019_23841	Bacteria(100);Firmicutes(100);Clostridia(100);Lachnospirales(100);Lachnospiraceae(100);Lachnospiraceae_NK4A136_group(94);
M00967_43_000000000-A3JHG_1_2101_16841_5683	Bacteria(100);Firmicutes(100);Clostridia(100);Lachnospirales(100);Lachnospiraceae(100);ASF356(100);
M00967_43_000000000-A3JHG_1_1113_12062_8139	Bacteria(100);Firmicutes(100);Clostridia(100);Oscillospirales(100);Ruminococcaceae(100);Incertae_Sedis(97);
M00967_43_000000000-A3JHG_1_2103_24284_21559	Bacteria(100);Firmicutes(100);Clostridia(100);Lachnospirales(100);Lachnospiraceae(100);Lachnospiraceae_unclassified(100);
M00967_43_000000000-A3JHG_1_2109_25056_22893	Bacteria(100);Bacteroidota(100);Bacteroidia(100);Bacteroidales(100);Muribaculaceae(100);Muribaculaceae_ge(98);
M00967_43_000000000-A3JHG_1_2113_12029_24665	Bacteria(100);Firmicutes(90);Clostridia(89);Clostridia_vadinBB60_group(87);Clostridia_vadinBB60_group_fa(87);Clostridia_vadinBB60_group_ge(87);
M00967_43_000000000-A3JHG_1_2110_11085_20271	Bacteria(100);Firmicutes(100);Clostridia(100);Clostridiales(100);Clostridiaceae(100);Clostridium_sensu_stricto_1(86);
```

- classification-summary.tsv = the number of sequences that were found at each level



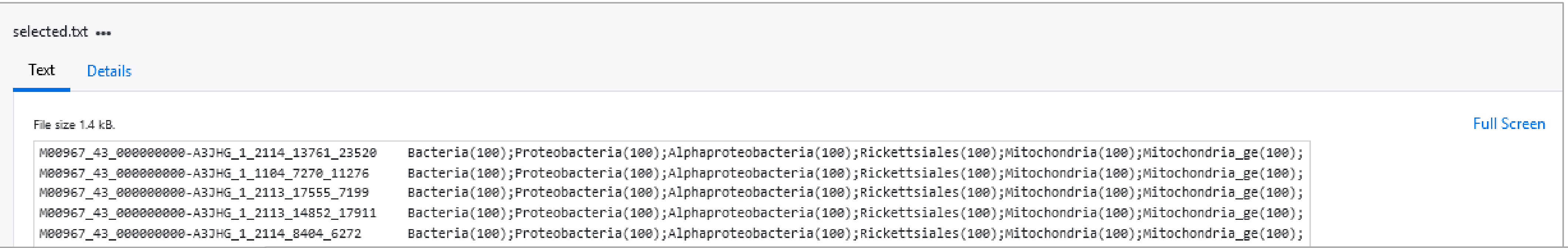
classification-summary.tsv •••

Showing the first 100 of 208 rows. View in full screen to see all rows.

taxlevel	rankID	taxon	daughterlevels	total	F3D0	F3D1	F3D141	F3D142	F3D143	F3D144	F3D145	F3D146	F3D147	F3D148	F3D149	F3D150	F3D2	F3D3	F3
0	0	Root	1	123474	6573	4906	5046	2634	2636	3842	6063	4246	14014	10415	10962	4558	16667	5628	368
1	0.1	Bacteria	9	123474	6573	4906	5046	2634	2636	3842	6063	4246	14014	10415	10962	4558	16667	5628	368
2	0.1.1	Actinobacteriota	2	400	27	4	20	29	13	28	10	9	50	81	43	28	19	20	3
3	0.1.1.1	Actinobacteria	3	265	23	0	16	28	10	21	7	3	29	68	25	16	4	15	0
4	0.1.1.1.1	Actinomycetales	1	2	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0
5	0.1.1.1.1.1	Actinomycetaceae	1	2	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0
6	0.1.1.1.1.1.1	Actinomyces	0	2	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0
4	0.1.1.1.2	Bifidobacteriales	1	262	23	0	16	27	10	21	5	3	29	68	25	16	4	15	0
5	0.1.1.1.2.1	Bifidobacteriaceae	2	262	23	0	16	27	10	21	5	3	29	68	25	16	4	15	0

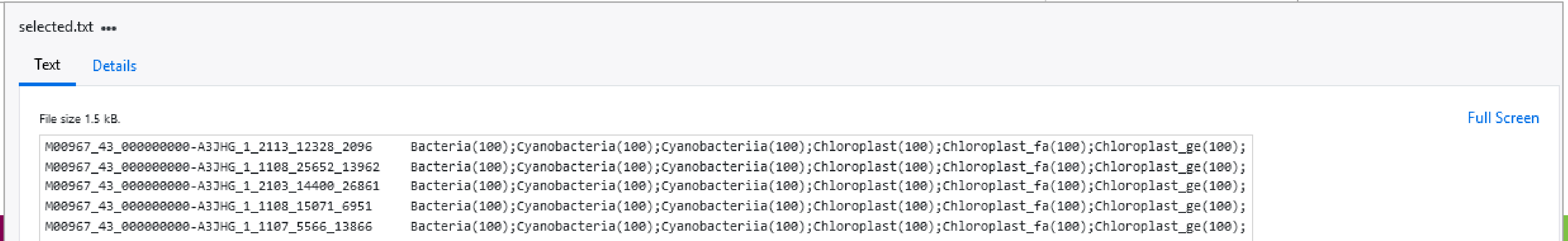
Removing unwanted lineages

- Data may contain assignments to mitochondria, chloroplasts, unknown
- You can remove these after converting Mothur files into phyloseq object
  - Remove selected taxa removes chloroplast and mitochondrial sequences from a phyloseq object, and up to five user-specified taxa at the desired level of biological organization.
  - Filter by taxonomic group tidies a phyloseq object so that OTUs only from the desired taxonomic group (bacteria, archaea, eukaryotes or fungi) are retained. Features with ambiguous phylum-level annotation (e.g. NA, unknown, uncharacterized) are removed.



File size 1.4 kB.

```
M00967_43_000000000-A3JHG_1_2114_13761_23520    Bacteria(100);Proteobacteria(100);Alphaproteobacteria(100);Rickettsiales(100);Mitochondria(100);Mitochondria_ge(100);
M00967_43_000000000-A3JHG_1_1104_7270_11276     Bacteria(100);Proteobacteria(100);Alphaproteobacteria(100);Rickettsiales(100);Mitochondria(100);Mitochondria_ge(100);
M00967_43_000000000-A3JHG_1_2113_17555_7199     Bacteria(100);Proteobacteria(100);Alphaproteobacteria(100);Rickettsiales(100);Mitochondria(100);Mitochondria_ge(100);
M00967_43_000000000-A3JHG_1_2113_14852_17911    Bacteria(100);Proteobacteria(100);Alphaproteobacteria(100);Rickettsiales(100);Mitochondria(100);Mitochondria_ge(100);
M00967_43_000000000-A3JHG_1_2114_8404_6272      Bacteria(100);Proteobacteria(100);Alphaproteobacteria(100);Rickettsiales(100);Mitochondria(100);Mitochondria_ge(100);
```



File size 1.5 kB.

```
M00967_43_000000000-A3JHG_1_2113_12328_2096     Bacteria(100);Cyanobacteria(100);Cyanobacteriia(100);Chloroplast(100);Chloroplast_fa(100);Chloroplast_ge(100);
M00967_43_000000000-A3JHG_1_1108_25652_13962    Bacteria(100);Cyanobacteria(100);Cyanobacteriia(100);Chloroplast(100);Chloroplast_fa(100);Chloroplast_ge(100);
M00967_43_000000000-A3JHG_1_2103_14400_26861    Bacteria(100);Cyanobacteria(100);Cyanobacteriia(100);Chloroplast(100);Chloroplast_fa(100);Chloroplast_ge(100);
M00967_43_000000000-A3JHG_1_1108_15071_6951     Bacteria(100);Cyanobacteria(100);Cyanobacteriia(100);Chloroplast(100);Chloroplast_fa(100);Chloroplast_ge(100);
M00967_43_000000000-A3JHG_1_1107_5566_13866     Bacteria(100);Cyanobacteria(100);Cyanobacteriia(100);Chloroplast(100);Chloroplast_fa(100);Chloroplast_ge(100);
```