ONTbarcoder

A pipeline for MinION based DNA barcoding

ONTBarcoder (available from https://github.com/asrivathsan/ONTbarcoder) is installed by unzipping the folder with the version of the program that supports the operating system on your computer. The folders are available from releases in Github. . For MacOS, an .app bundle has been created. Kindly download the bundle relevant to the OS version. For further notes on MacOS compatibility and permissions: please see Page 13. Currently the software has been tested in OSX 11.4, 10.13, and 10.12.

ONTBarcoder is designed to handle dual tagged amplicon pools. It performs demultiplexing and consensus calling If the software is used for the whole process, the format specifications are relatively simple and you can skip to the next section. If however, the demultiplexing is carried out independently, please follow the guidelines in Format Specifications, Point 2 in Page 12 of this documentation. It would be important to ensure that consensus calling is done under the assumption that primers and tags are removed. At least, tags should be removed if primers do not have ambiguous bases. The length parameters should be modified to expected consensus length.

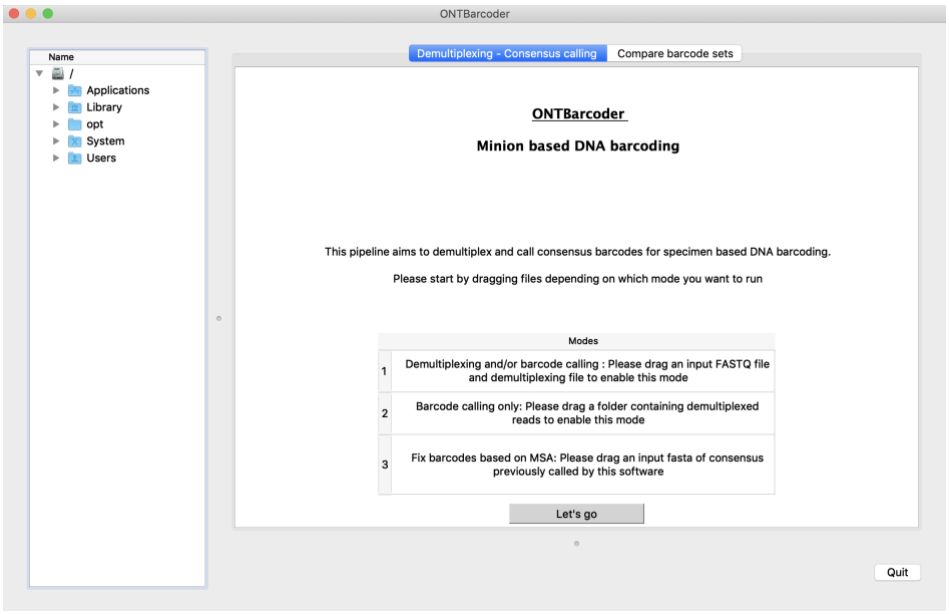
ONTBarcoder's three modules.

1. Demultiplexing: MinION reads are assigned to specimen-specific bins.
2. Barcode calling: The reads in the specimen-specific bins are used to derive the barcodes based on alignment and consensus calling.
3. Barcode comparison module: Two or more sets of barcodes can be compared.

We recommend that you try the pipeline using our DatasetA containing 257 amplicons generated from Flongle. It is available from here. Information to download the various datasets is available here. Prior to running any dataset, ensure you have at least 1.5 times the space required for the input FASTQ file.

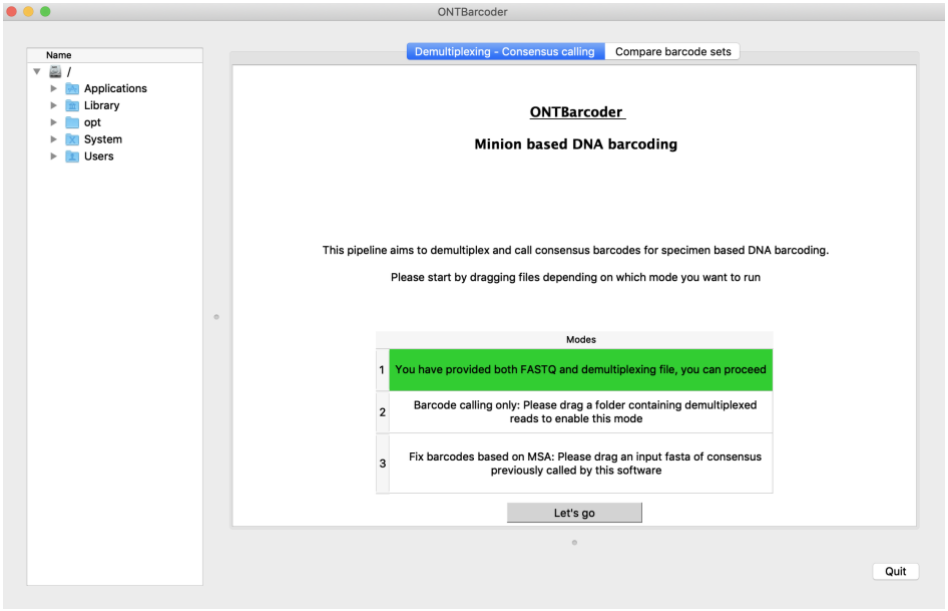
A simple video tutorial is available.

1. Open ONTbarcoder



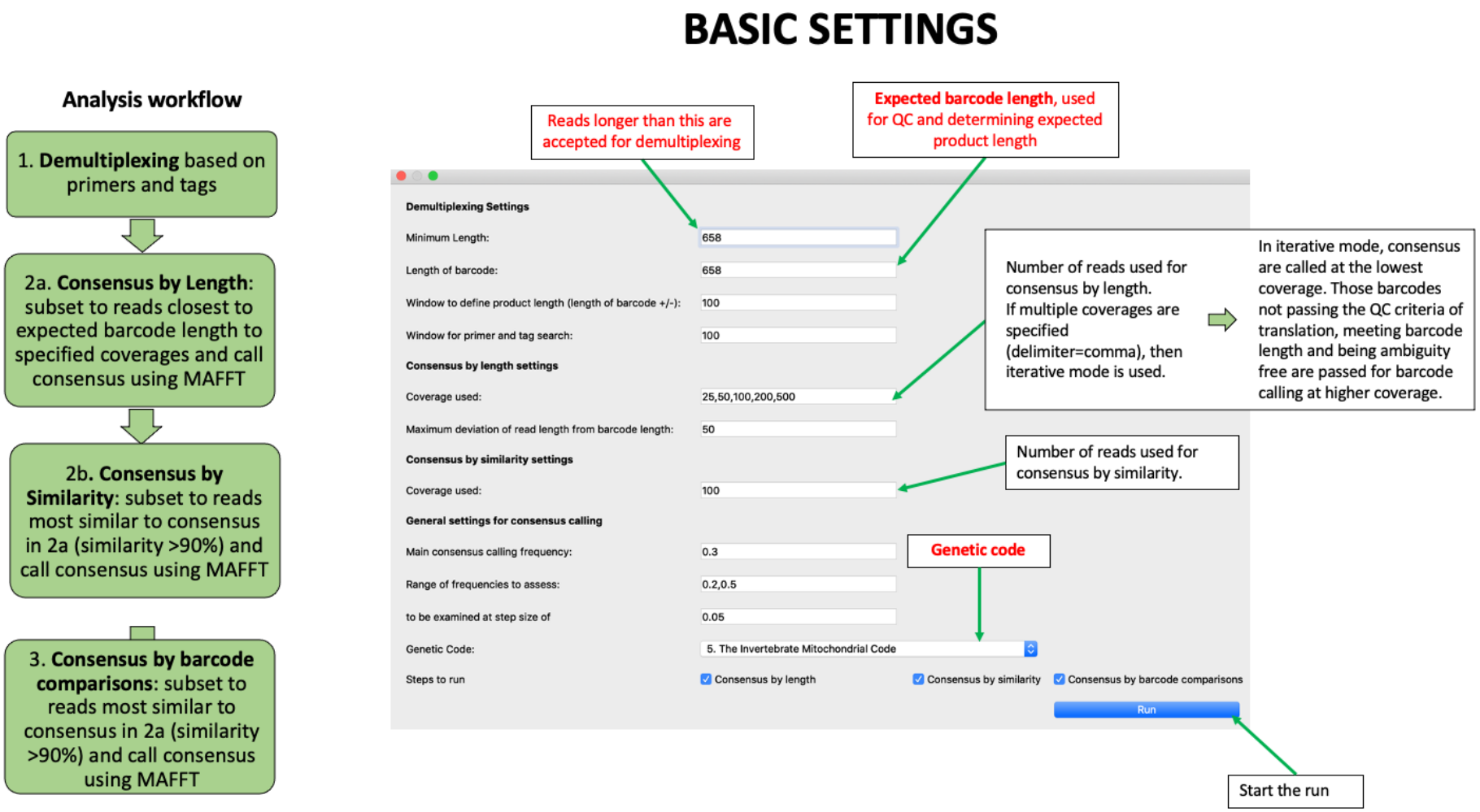
Demultiplexing and barcode calling using the "Demultiplexing-Consensus calling tab"

2. Drag in Input Fastq and demultiplexing file, either simultaneously or one by one

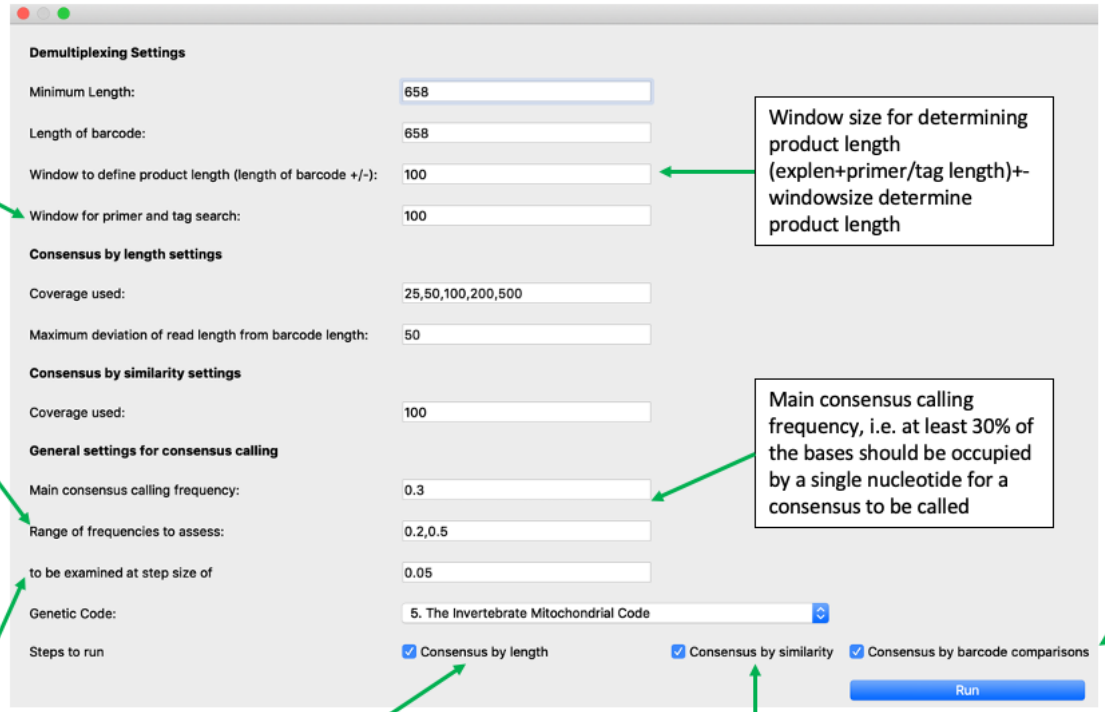


3. Press “Let’s go”

4. Configure your run based on the settings described below. DatasetA does not require any changes to the default. The default assumes that you are working on invertebrate barcodes that are 658 bp in length. The remaining settings define which reads are demultiplexed, where the primers are found, how many reads are used in the different steps, and how the consensus barcodes are determined.



ADVANCED SETTINGS



Primers & tags are searched within the first/last specified number of bp of a read

Window size for determining product length (explen+primer/tag length)+- windowsize determine product length

Other consensus calling frequencies to test

Main consensus calling frequency, i.e. at least 30% of the bases should be occupied by a single nucleotide for a consensus to be called

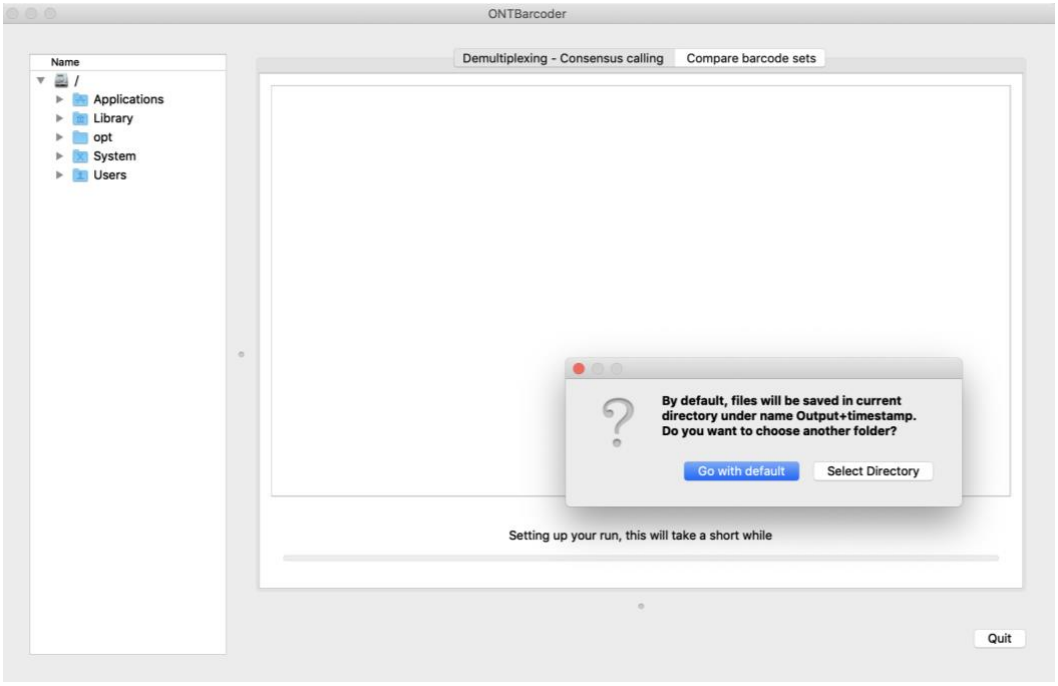
Specifies the steps for the range. In current settings 0.2,0.25,0.3,0.35,0.4,0.45,0.5 are tested

Enables/disables **Consensus by barcode comparisons,** i.e. barcode fixing that fills/ eliminates gaps in barcodes by comparison to other barcodes in the run

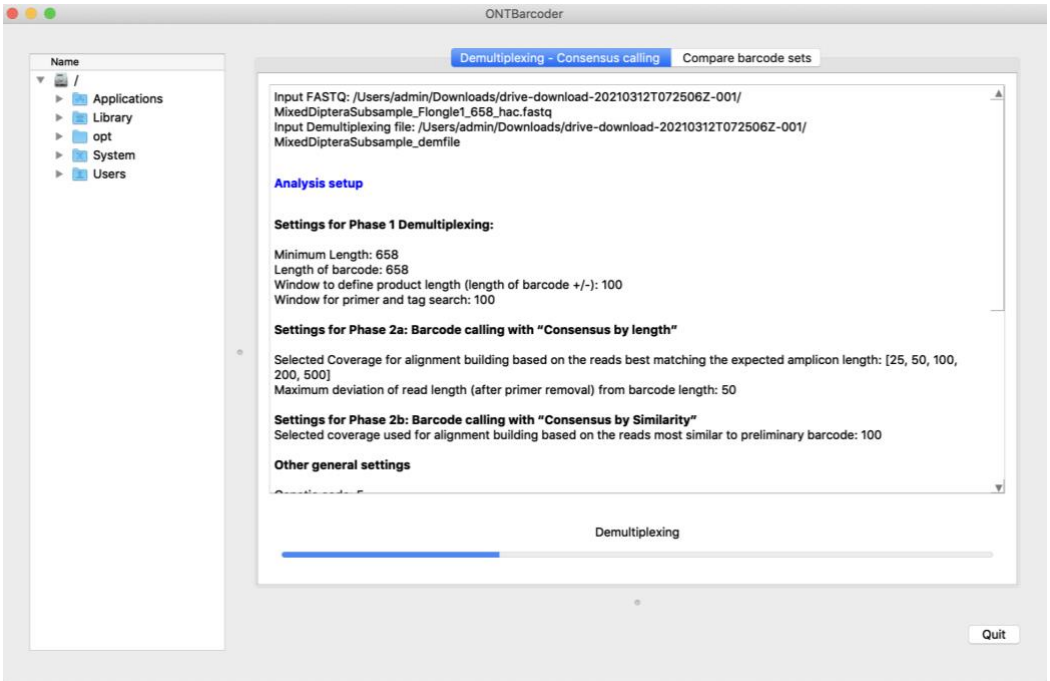
Enables/disables **Consensus by length**: Step that uses reads closest to the known length of the barcode are used for consensus calling.

Enables/disables **Consensus by similarity**: Step that improves consensus in previous step by taking reads most similar to it.

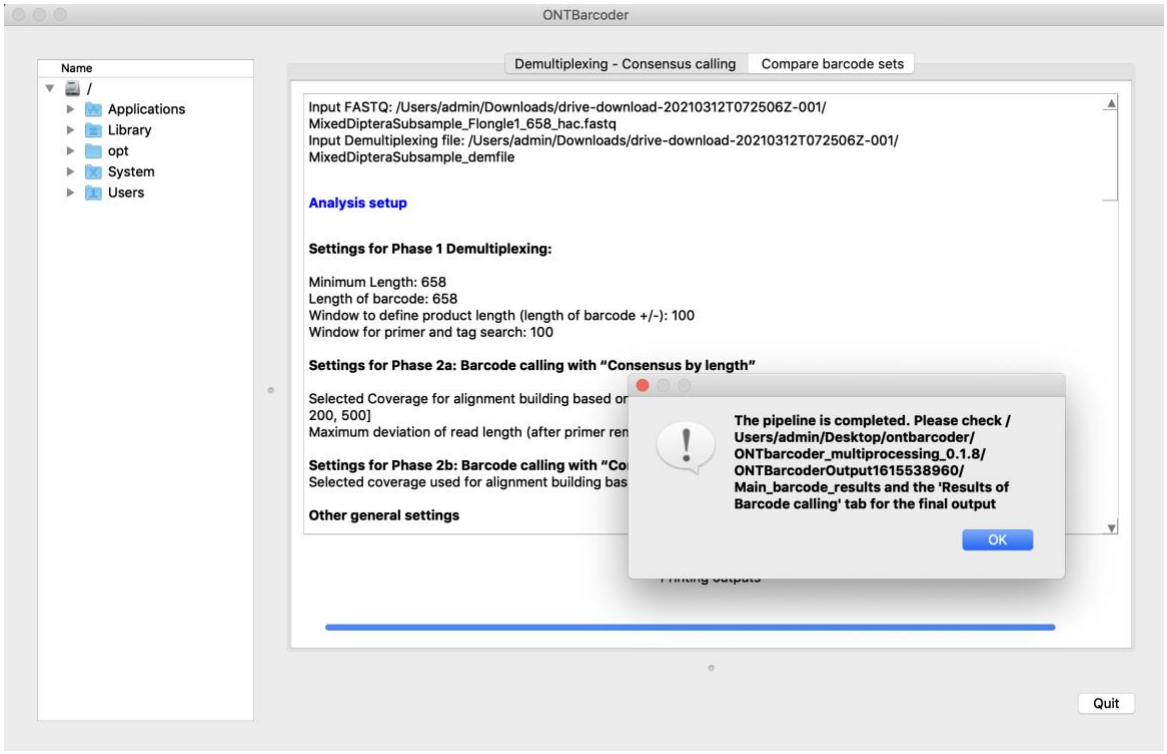
5. Select an output folder. If you select “default”, it will create a folder within the directory containing the software with a unique name starting with ONTbarcoderOutput. If you choose “Select Directory” ensure that the directory name has no empty space and is empty. Generally avoid complex characters in your names (spaces, brackets, slashes, colons)



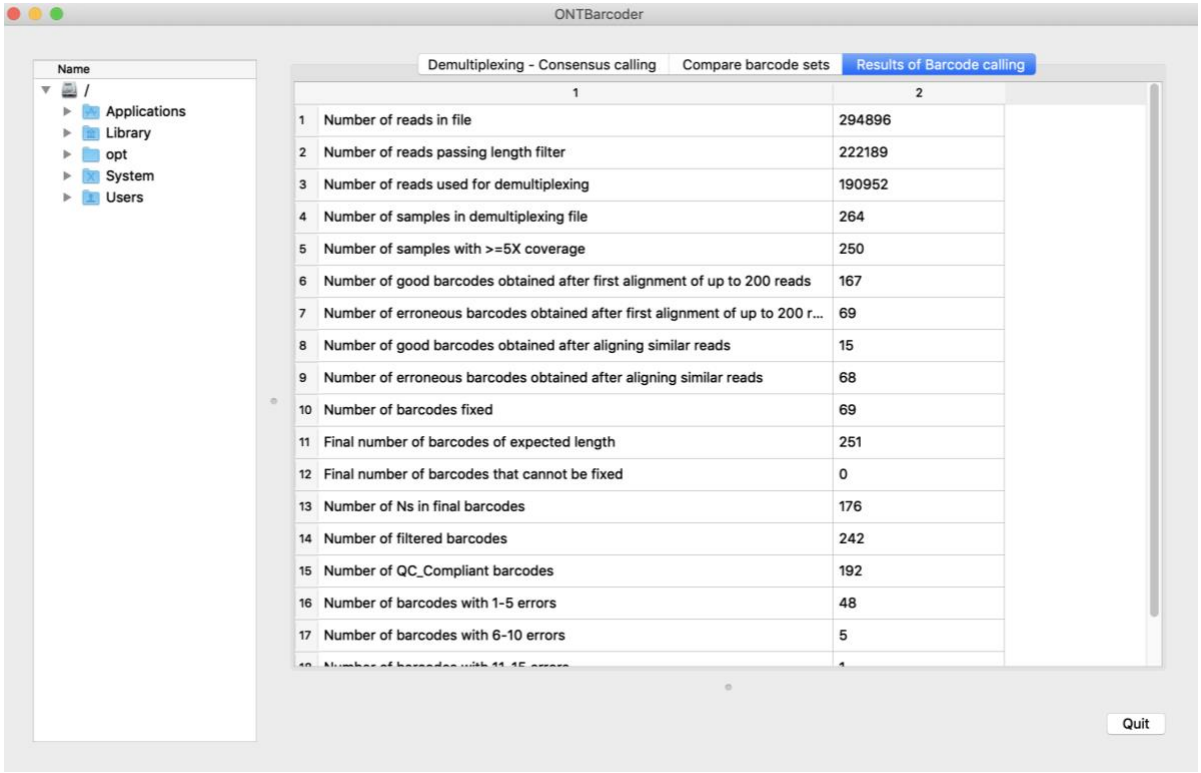
6. The analysis starts, a progress bar appears, and results are generated in respective folders as the analysis advances.



7. The analysis is complete and all output files have been created.

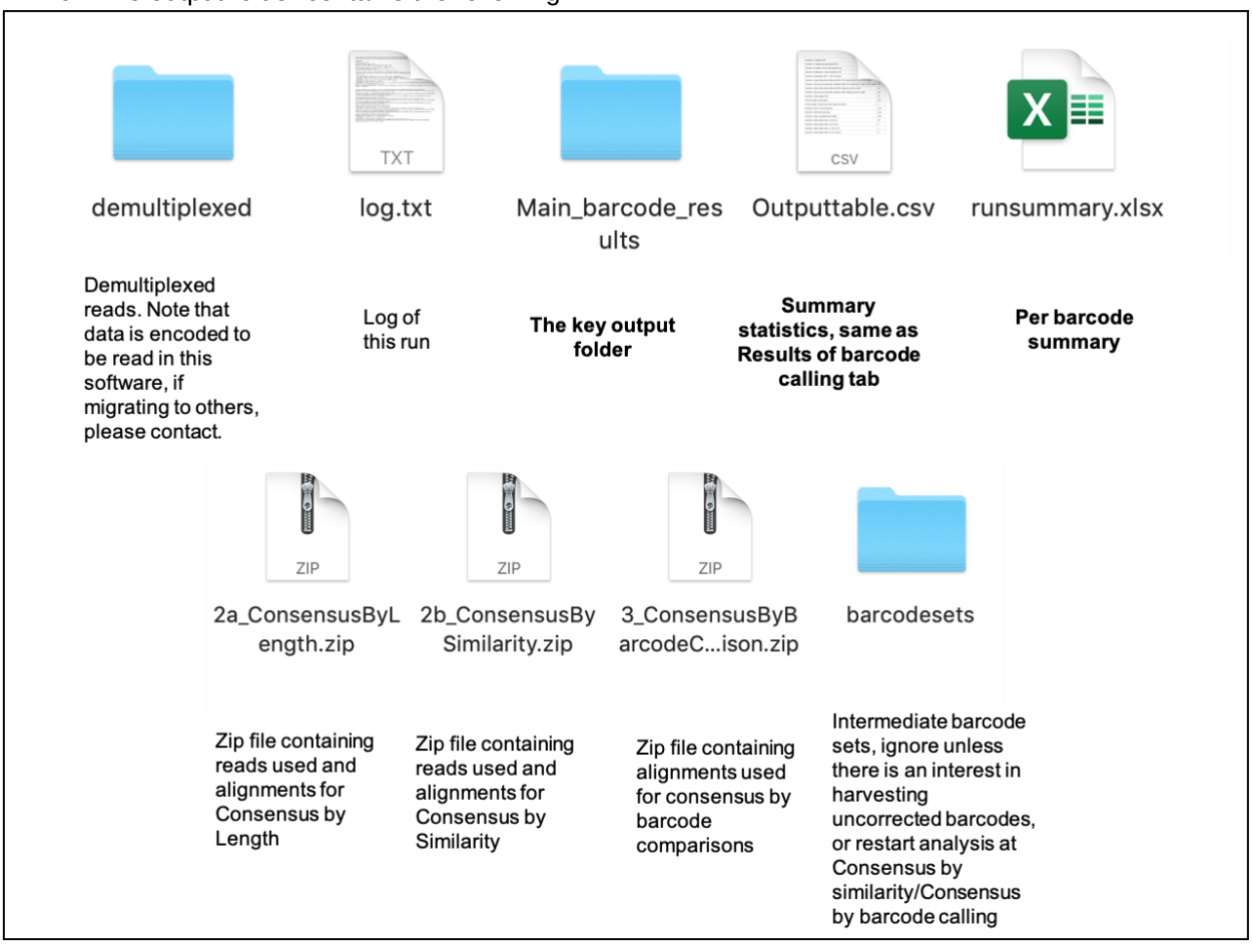


8. Results Table: The table reports that 242 "Filtered" barcodes were obtained of which 192 are QC-Compliant. This means they have no ambiguities, are translatable, have expected barcode length and are not having any gaps in an internal Multiple Sequence Alignment (MSA) check. The filtered sets include addition 50 barcodes that have <1% ambiguous bases, are translatable, have expected length, and may have been corrected for up to 5 indels. The table can be copied but it also stored in the "Outputtable.csv" file in the output folder.

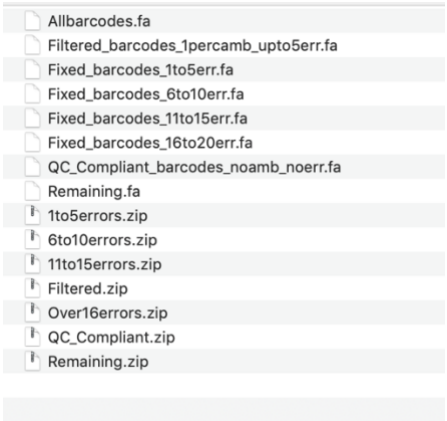


	1	2
1	Number of reads in file	294896
2	Number of reads passing length filter	222189
3	Number of reads used for demultiplexing	190952
4	Number of samples in demultiplexing file	264
5	Number of samples with >=5X coverage	250
6	Number of good barcodes obtained after first alignment of up to 200 reads	167
7	Number of erroneous barcodes obtained after first alignment of up to 200 r...	69
8	Number of good barcodes obtained after aligning similar reads	15
9	Number of erroneous barcodes obtained after aligning similar reads	68
10	Number of barcodes fixed	69
11	Final number of barcodes of expected length	251
12	Final number of barcodes that cannot be fixed	0
13	Number of Ns in final barcodes	176
14	Number of filtered barcodes	242
15	Number of QC_Compliant barcodes	192
16	Number of barcodes with 1-5 errors	48
17	Number of barcodes with 6-10 errors	5

9. The output folder contains the following



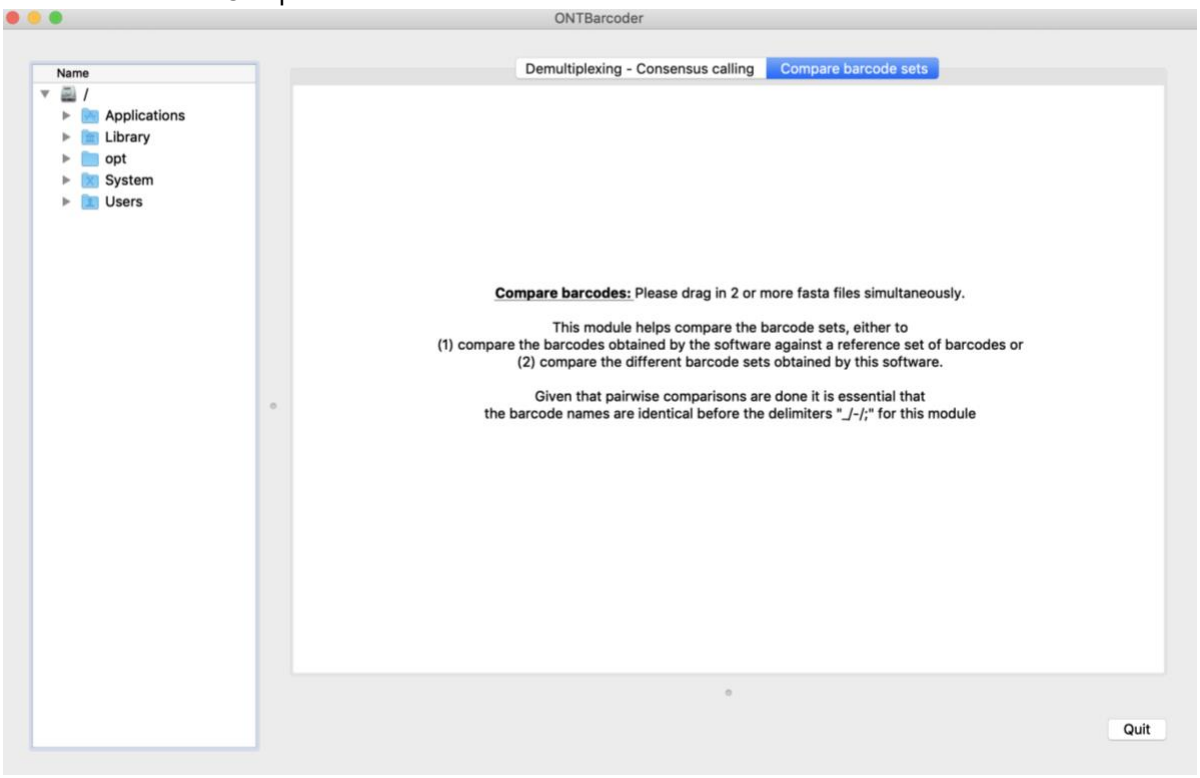
10. The Main Barcode Results folder contains the main barcode output file **"Filtered_barcodes"** fasta file. The barcodes without ambiguous bases are in the QC_Compliant_barcodes file. The zip folders contain demultiplexed reads per dataset divided into each of the categories. All barcodes should be checked via BLAST for contamination and/or verification that the barcode obtained belongs to the expected taxon.



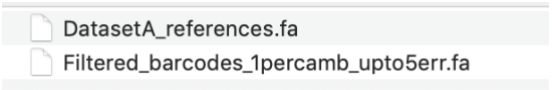
2. Compare barcodes in the “Compare barcodes sets tab”

This can be used in different modes (4 modes: single barcode file against single reference, multiple barcode files against single reference, pairwise comparisons of two barcode files, and all-vs-all comparisons of multiple barcode files), which depend on how many input files are dragged in. Here we show how to compare the output for Dataset A with reference sequences generated with Sanger sequencing.

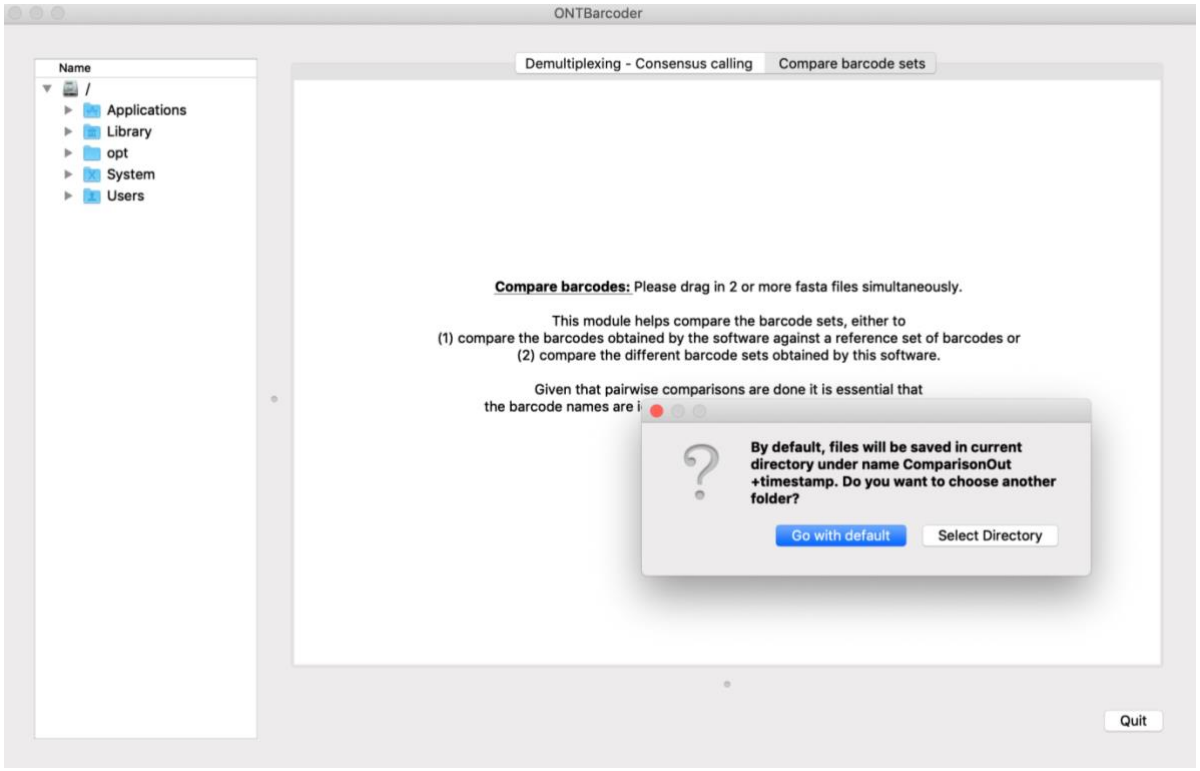
1. Switch to “Compare barcode sets” tab



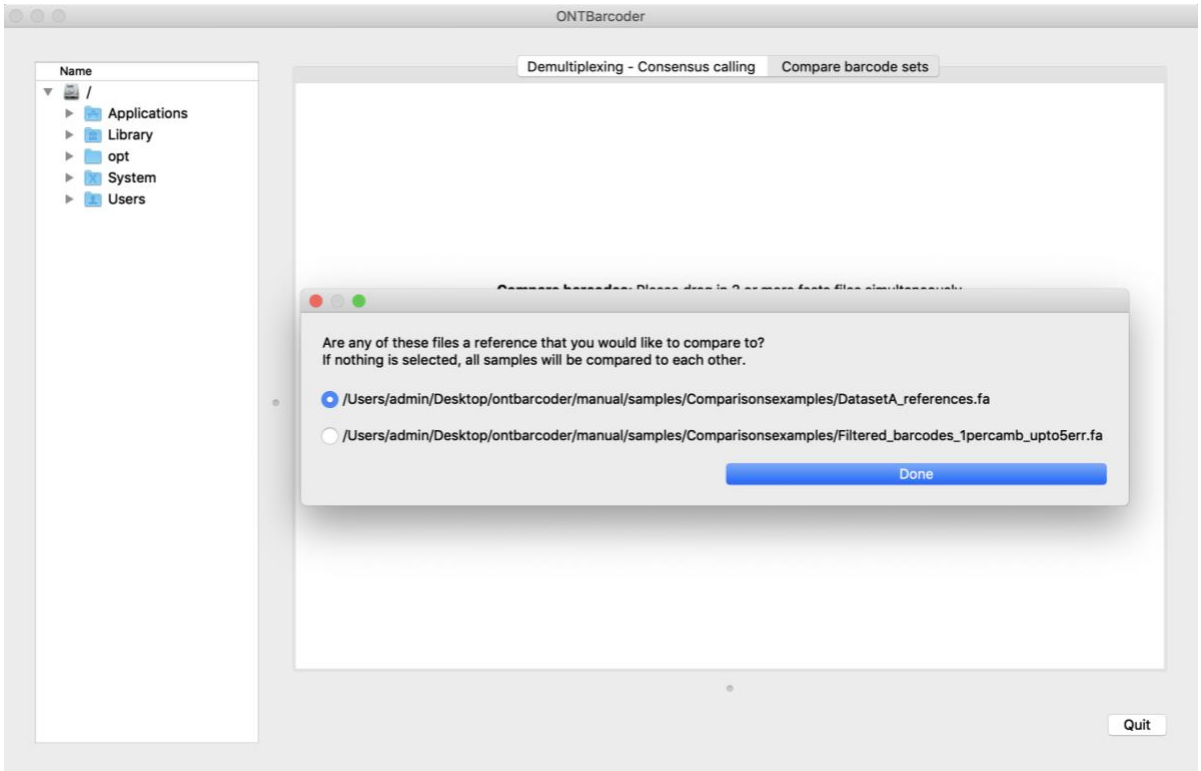
2. Drag at least two fasta files into the window. For the example, go to the “Main_barcode_results” folder and use the following two files:



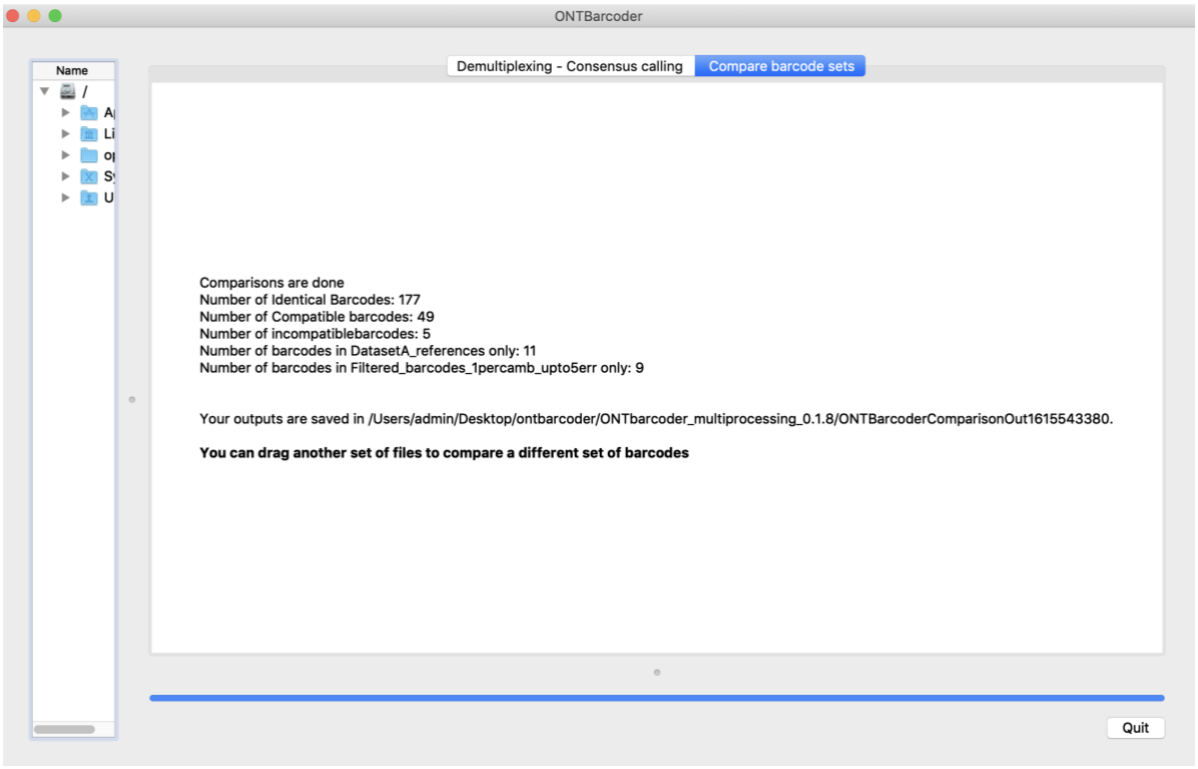
3. Select an output mode, if you select default, it will create a folder within the directory containing the software starting with ONTBarcoderComparisonOut. If you choose “Select Directory” ensure that the directory name has no empty space and is empty. Generally avoid complex characters in your names (space, brackets, slashes, colons)



4. You can now select a set of reference barcodes if you want all other files to be compared to the barcodes in the reference set. Alternatively, you can just press “Done” to do all pairwise comparisons between the fasta files. Here, since we want the reference mode, we select the reference Sanger file and click “Done”

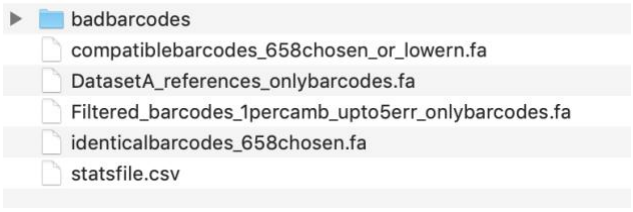


5. The output will be as shown below and the text can be copied. It is also in the output folder

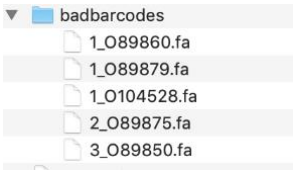


The barcodes in the sets are compared and classified into three categories: “identical” where sequences are a 100% match and lack ambiguities, “compatible” where the sequences only differ by ambiguities, and “incompatible” where the sequences differ by at least one base pair.

6. The output folder contains the following:



- “statsfile.csv” is the key summary folder having summary of the comparisons and the various incompatible barcodes with the edit distances
- Identicalbarcodes* and compatible* barcode files gives one sequence per dataset, in case the user wants to dereplicate barcodes
- The *onlybarcodes.fa files contain barcodes that cant be compared
- “badbarcodes” folder contains incompatible barcodes and the value before “_” represents edit distance.



Note that if more than 2 input files were provided, each pairwise comparison is given in a separate folder and a summary of all comparisons is then given the outermost folder

Summary

Expected input and output

1. Demultiplexing module	Input required	Results
Demultiplexing	• FASTQ file obtained after base-calling • Demultiplexing file	• Demultiplexed reads in "demultiplexed" folder • Overall summary in "Outputtable.csv" • Per barcode summary in "runsummary.xlsx
2. Barcode calling module	**Input required**	**Results**
Barcodes are derived from reads in specimen-specific bins	• Folder containing Demultiplexed FASTA files. Format specified under "Directory containing Demultiplexed FASTA files"	• Barcode sets in "Main_barcode_results" folder • Overall summary in "Outputtable.csv" • Per barcode summary in "runsummary.xlsx"
Improving barcodes using "Consensus by similarity" and "Consensus by barcode comparisons"	• Sequences in "barcodesets" folder	• Barcode sets available from "Main_barcode_results" folder • Overall summary in "Outputtable.csv" • Per barcode summary in "runsummary.xlsx"
3. Barcode comparison module	**Input required**	**Results**
Comparison of barcodes to references	• One or more barcode fasta file(s) • Reference fasta file *See format specifications*	• "statsfile.csv" that describes the overall summary and erroneous barcodes
Comparison of barcodes to each other	• Two or more barcode fasta file(s) *See format specifications*	• "statsfile.csv" that describes the overall summary and erroneous barcodes

Format Specifications

File	
FASTQ file	Standard fastq, generated after basecalling with ONT software
1. Demultiplexing File	A 5-column csv file with the following headers: SpecimenID, TagFsequence, TagRsequence, PrimerF, PrimerR You can only demultiplex one one primer pair at a time. FASTQ files with data for multiple pairs, have to processed sequentially. **Please avoid unusual characters in Specimen ID (e.g. characters like "(){}[]\/.,;*$" will lead to crashes)**
2. Directory containing Demultiplexed FASTA files	Input for Mode 2 should be a directory that contains only FASTA files. Directory should not be empty nor should it contain other files. The fasta files must have names as "sampleID_all.fa" i.e. the suffix _all.fa is critical
3. Files for comparison module	1. Specimen barcodes in different files should have identical name before the following delimiter characters ("_/-/;") 2. The barcodes for at least some of the specimens should be present in multiple input files
4. Input file for improvement (Step 4)	ONTbarcoder's pipeline can be started at different points. Demultiplexing files can be used to only carry out barcode calling.

	Consensus by barcode can be started with files that have the format specified in the hyperlink

NOTES on MacOS installation.

Apple made a recent upgrade to Big Sur which has led to many software are facing compatibility issues. Currently we have tested this Big Sur 11.4 and it has worked smoothly. However, it was noted that permissions can be an issue in Mac. Transferring the app to Applications folder and running worked smoothly. If however one faces permission issues please modify System Preferences > Security and Privacy > Full Disk Access.

A separate compilation was made in OSX 10.13.6 High Sierra and this version of the software is compatible with Sierra 10.12

In order to trouble issues in mac, one may want to run the software from Terminal, under and move in the the /path-to-.app/Contents/MacOS/ and run the ONTbarcoder executable as ./ONTbarcoder_multiprocessing.