Script P1: Pre-Processing Samples

Hannigan GD, Grice EA, et al.

Abstract

This protocol provides methods for quality control of metagenomic data. Included is adapter trimming, quality trimming, decontamination, negative control removal, and pre-processing results. Based on the methods found in the following publication:

Hannigan, Geoffrey D., et al. "The Human Skin Double-Stranded DNA Virome: Topographical and Temporal Diversity, Genetic Enrichment, and Dynamic Associations with the Host Microbiome." *mBio* 6.5 (2015): e01578-15.

Citation: Hannigan GD, Grice EA, et al. Script P1: Pre-Processing Samples. **protocols.io** dx.doi.org/10.17504/protocols.io.edrba56
Published: 10 Mar 2016

Guidelines

Required Software:

- cutadapt-1.4.1
- fastx_toolkit-0.0.14
- NCBI's BLAST+ v2.2.0
- bwa-0.5.9
- deconseq-standalone-0.4.3

Relevant Files

Output:

- Virome_Sequence_Counts
- Whole_Microbiome_Sequence_Counts

Perl Scripts:

- calculate_fasta_median_length.pl
- calculate_fastq_median_length.pl

R Scripts:

- R1, R2, and R3
- human_decontamination_stats.R

Before start

Perl scripts and other supplementary information available at:

https://figshare.com/articles/The_Human_Skin_dsDNA_Virome_Topographical_and_Temporal_Diversity_Genetic_Enrichment_and_Dynamic_Associations_with_the_Host_Microbiome/1281248

Protocol

Adapter Trimming

Step 1.

Make directory for the output.

cmd COMMAND

```
mkdir ./fastq_trimmed_adapters
mkdir ./fastq_trimmed_adapters_STDERR
```

Adapter Trimming

Step 2.

Write function to trim adapters.

SOFTWARE PACKAGE (Unix)

cutadapt, 1.4.1

Marcel Martin
https://github.com/marcelm/cutadapt/tree/25fa1e828b0f737dc43b7aec9f29582fbcb0245e/cutadapt

cmd COMMAND

```
runCutadaptWithMap() {
    echo Input fastq = ${1} #1 = full path to fastq file to be trimmed
    echo Mapping file = ${2} #2 = full path to mapping file mapping file
    echo Output file = ${3} #3 = full path for output directory
    echo STDERR file = ${4} #4 = full path to sterr result directory
    export SAMPLEID=$(echo ${1} | sed 's/^.*\///g' | sed 's/_R.*$//')
    export THREEPRIME=$(awk --
assign sampleid=$SAMPLEID '$1 == sampleid { print $13 }' ${2})
    export FIVEPRIME=$(awk --assign sampleid=$SAMPLEID '$1 == sampleid { print $12 }' ${2})
    /data/apps/bin/cutadapt --error-rate=0.1 --overlap=10 -a $THREEPRIME -
a $FIVEPRIME ${1} > ${3} 2> ${4}
    gzip ${3}
}
```

This function will remove adapters with an error rate of 0.1 and overlap of 10.

Adapter Trimming

Step 3.

Set the function to be used in loops.

cmd COMMAND

```
export -f runCutadaptWithMap
```

Adapter Trimming

Step 4.

Trim the adapters from the fastq files of interest.

cmd COMMAND

```
ls ./raw_fastq/ | xargs -I {} --max-procs=8 sh -
c 'runCutadaptWithMap ./raw_fastq/{} SkinMet_Virome_metadata.tsv ./fastq_trimmed_adapters/{
} ./fastq_trimmed_adapters_STDERR/{}'
```

raw_fastq is the directory name where the fastq files are found. Rename as needed.

Quality Trimming

Step 5.

Remove sequences with a quality score < 33 from the adapter trimmed sequences using the fastx toolkit.

SOFTWARE PACKAGE (Unix)

FASTX Toolkit, 0.0.14

Hannon Lab
https://github.com/agordon/fastx_toolkit

cmd COMMAND

```
mkdir ./trimmed_fastq/
ls ./fastq_trimmed_adapters/* | sed -e 's/^.*\/.*\///g' | xargs -I {} --max-
procs=18 fastq_quality_trimmer -t 33 -Q 33 -i ./fastq_trimmed_adapters/{} -
o ./trimmed_fastq/{}
```

Decontamination

Step 6.

Remove sequencing mapping to the human genome using the DeconSeq toolkit with default parameters and the human reference GRCh37.

SOFTWARE PACKAGE (Unix)

DeconSeq toolkit, 0.4.3

Schmieder R and Edwards R
http://sourceforge.net/projects/deconseq/files/

cmd COMMAND

```
mkdir ./deconseq_fastq
ls ./trimmed_fastq/* | sed -e 's/^.*\/.*\///g' | xargs -I {} --max-procs=128 perl deconseq-
standalone-0.4.3/deconseq.pl -f ./trimmed_fastq/${} -dbs hsref -
out_dir ./deconseq_fastq/${}
wait
```

Decontamination

Step 7.

Automatically rename the output files so that they are specific for each sample.

cmd COMMAND

```
mkdir ./clean_fastq
mkdir ./cont_fastq

ls -d ./deconseq_fastq/* | sed 's/^.*\///g' | xargs -I {} --max-procs=1 sh -
c 'cp ./deconseq_fastq/{}/*_clean.fq ./clean_fastq/{}'
ls -d ./deconseq_fastq/* | sed 's/^.*\///g' | xargs -I {} --max-procs=1 sh -
c 'cp ./deconseq_fastq/{}/*_cont.fq ./cont_fastq/{}'
```

Here we only rename the 'clean' files because those are the ones we will continue to work with downstream.

Decontamination

Step 8.

Since a 1% spike-in of PhiX Control was added to the Illumina sequencing runs for quality control purposes, any sequences mapping to the PhiX174 genome (NCBI Accession: NC_001422) are also removed using DeconSeq, but only for the whole metagenome samples (virome samples were subjected to further background removal).

Decontamination

Step 9.

Build PhiX database in deconseq-standalone-0.4.3/db

cmd COMMAND

```
bwa-0.5.9/bwa64 index –p PhiX174 PhiX174.fasta
```

Decontamination

Step 10.
Make directory for PhiX decontaminated output.

cmd COMMAND
```
mkdir ./deconseq_phix_fastq
ls ./clean_fastq/* | sed -e 's/^.*\/.*\///g' | xargs -I {} --max-procs=128 perl deconseq-
standalone-0.4.3/deconseq.pl -f ./clean_fastq/{} -dbs phix -
out_dir ./deconseq_phix_fastq/{}
wait
```

Decontamination

Step 11.
Automatically rename the output files so that they are specific for each sample.

cmd COMMAND
```
mkdir ./clean_phix_fastq
mkdir ./cont_phix_fastq

ls -d ./deconseq_phix_fastq/* | sed 's/^.*\///g' | xargs -I {} --max-procs=1 sh -
c 'cp ./deconseq_phix_fastq/{}/*_clean.fq ./clean_phix_fastq/{}'
ls -d ./deconseq_phix_fastq/* | sed 's/^.*\///g' | xargs -I {} --max-procs=1 sh -
c 'cp ./deconseq_phix_fastq/{}/*_cont.fq ./cont_phix_fastq/{}'
```
Here we only rename the 'clean' files because those are the ones we will continue to work with downstream.

Negative Control Removal- Virome

Step 12.
Make a new directory for fasta files to be cleaned of environmental contamination.

cmd COMMAND
```
mkdir ./clean_R1_fasta
```

Negative Control Removal- Virome

Step 13.
Convert the deconseq cleaned fastq files to fasta files and deposit the sequence files into the new directory.

cmd COMMAND
```
for file in $(ls clean_phix_fastq/); do
    idba_ud-1.0.9/bin/fq2fa ./clean_phix_fastq/$file ./clean_R1_fasta/${file}
done
```

Negative Control Removal- Virome

Step 14.
Change the file extension names.

cmd COMMAND
```
for file in $(ls clean_R1_fasta/); do
        mv ./clean_R1_fasta/"${file}" ./clean_R1_fasta/"${file/%.fastq/.fa}"
done
```

Negative Control Removal- Virome

Step 15.
Remove the R2 (forward) files since this script is only cleaning the R1 direction reads.

cmd COMMAND
```
rm ./clean_R1_fasta/*R2*
```

Negative Control Removal- Virome

Step 16.
Format the fasta files to include the pair numbers information (1 or 2).

cmd COMMAND
```
mkdir ./clean_R1_fasta_neg_clean_format
```

```
for file in $(ls clean_R1_fasta/); do
        sed s'/ /\@/g' ./clean_R1_fasta/$file > ./clean_R1_fasta_neg_clean_format/${file}
done
```

Negative Control Removal- Virome

Step 17.

The formatted fasta files will be blasted against the NCBI non-redundant database with an evalue of 0.001 and only the top hit will. First create a directory for the blast results.

cmd COMMAND

```
mkdir ./blastn_nt_sequences
```

Negative Control Removal- Virome

Step 18.

Blast the sequences to use downstream for negative control removal.

SOFTWARE PACKAGE (Unix)

BLAST Toolkit, 2.2.0

NCBI
ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST/

cmd COMMAND

```
run.blastn.for.neg.cleaning.against.nt () {
    blastn -query ./clean_R1_fasta_neg_clean_format/${1} -out ./blastn_nt_sequences/${1} -
db /references/ncbi/nt -outfmt 6 -num_threads 2 -max_target_seqs 1 -evalue 1e-3
}
export -f run.blastn.for.neg.cleaning.against.nt
ls clean_R1_fasta_neg_clean_format/ | xargs -I {} --max-procs=128 sh -
c 'run.blastn.for.neg.cleaning.against.nt {}'
wait
```

A simple way to parallelize this proccess on our server is to run it as a subroutine.

Negative Control Removal- Virome

Step 19.

Rename the file extensions to be .txt

cmd COMMAND

```
for file in $(ls blastn_nt_sequences/); do
        mv ./blastn_nt_sequences/"${file}" ./blastn_nt_sequences/"${file/%.fa/.txt}"
done
```

Negative Control Removal- Virome

Step 20.

Annotate the sequences with their corresponding gi numbers.

cmd COMMAND

```
mkdir ./tmp-dir-list1
mkdir ./blastn_nt_sequences_only
cp ./blastn_nt_sequences/MG*.txt ./blastn_nt_sequences_only
for file in $(ls ./blastn_nt_sequences_only); do
        sed 's/^.*gi|//' ./blastn_nt_sequences_only/$file | sed 's/|.*//'> ./tmp-dir-
list1/${file}_list1
done
```

Negative Control Removal- Virome

Step 21.

Make a list of the annotated sequence numbers.

cmd COMMAND

```
mkdir ./tmp-dir-list2
for file in $(ls ./blastn_nt_sequences_only); do
        cut -f 1 ./blastn_nt_sequences_only/$file > ./tmp-dir-list2/${file}_list2
done
```

Negative Control Removal- Virome

Step 22.
Merge the two lists from previous step together.

cmd COMMAND
```
mkdir tmp-dir-paired-lists
for i in $(ls ./tmp-dir-list1); do
for j in $(ls ./tmp-dir-list2); do
        for x in ${i::12}; do
        for y in ${j::12}; do
                if [ "$x" == "$y" ]; then
                paste ./tmp-dir-list2/$j ./tmp-dir-list1/$i > ./tmp-dir-paired-
lists/${i}_gi_reference
                fi
        done
        done
done
done
```

Negative Control Removal- Virome

Step 23.
Add > to the start of sequence names and merge gi number to the end of the sequence.

cmd COMMAND
```
mkdir ./tmp-dir-for-fa
for file in $(ls ./tmp-dir-paired-lists); do
    sed 's/^/>/' ./tmp-dir-paired-lists/$file | sed 's/\t/\@\@/' > ./tmp-dir-for-
fa/${file}_for_fa
done
```

Use double @ (@@) because the single @ was already used to designate a different separation above.

Negative Control Removal- Virome

Step 24.
Merge the list of sequence names and gi numbers to the list of un-merged sequence names.

cmd COMMAND
```
mkdir ./tmp-dir-paired-ids-with-gi
for i in $(ls ./tmp-dir-list2); do
for j in $(ls ./tmp-dir-for-fa); do
        for x in ${i::12}; do
        for y in ${j::12}; do
                if [ "$x" == "$y" ]; then
                paste ./tmp-dir-list2/$i ./tmp-dir-for-fa/$j > ./tmp-dir-paired-ids-with-
gi/${i}_gi_reference
                fi
        done
        done
done
done
```

Negative Control Removal- Virome

Step 25.
Add > to the start of sequence ids.

cmd COMMAND
```
mkdir ./tmp-dir-paired-ids-with-gi-for-fa
for file in $(ls ./tmp-dir-paired-ids-with-gi); do
        sed 's/^/>/' ./tmp-dir-paired-ids-with-gi/$file > ./tmp-dir-paired-ids-with-gi-for-
fa/${file}_for_fa
done
```

Negative Control Removal- Virome

Step 26.

Convert sequence numbers to gi ID number annotations.

cmd COMMAND

```
mkdir tmp-dir-gi-contigs
for i in $(ls ./tmp-dir-paired-ids-with-gi-for-fa); do
for j in $(ls ./clean_R1_fasta_neg_clean_format); do
        for x in ${i::12}; do
        for y in ${j::12}; do
                if [ "$x" == "$y" ]; then
                awk 'FNR==NR { a[$1]=$2; next } $1 in a { $1=a[$1] }1' ./tmp-dir-paired-
ids-with-gi-for-fa/$i ./clean_R1_fasta_neg_clean_format/$j > ./tmp-dir-gi-
contigs/${i}_gi_contigs
                fi
        done
        done
done
done
```

Negative Control Removal- Virome

Step 27.

Convert fasta files to single instead of double lined (this makes for easier removal of contig sequences using grep).

cmd COMMAND

```
mkdir ./tmp-dir-for-neg-removal
for file in $(ls ./tmp-dir-gi-contigs); do
        paste -s -d' \n' ./tmp-dir-gi-contigs/$file > ./tmp-dir-for-neg-removal/${file}
done
```

Negative Control Removal- Virome

Step 28.

In the original study, background control samples were collected from each subject at each sampling time point. The contaminants found in the control samples at a specific time point were removed from all other samples from the same subject at that time point. The samples will need to be seperated out by patient number so that the background control sequences for each specific patient can correspond to the other samples.

Negative Control Removal- Virome

Step 29.

Make tmp dir for cleaned fasta intermediate files.

cmd COMMAND

```
mkdir ./tmp-neg-clean
```

Negative Control Removal- Virome

Step 30.

Remove singletons from the GI reference list. First generate a list of gi numbers with line counts to detect the duplicates.

cmd COMMAND

```
mkdir ./tmp-dir-list1-no-singletons
for file in $(ls ./tmp-dir-list1); do
    sort ./tmp-dir-list1/$file | uniq -c | sed "s/^\s*//" | sed "s/\s/\t/" | grep -v -
P '^1\t.*' > ./tmp-dir-list1-no-singletons/${file}
done
wc -l ./tmp-dir-
list1/* | sed "s/\.\/.*1\///" | sed "s/\.txt.*$//" | sed "s/^\s*//" | sed "s/\s/\t/" > neg_
clean_with_singleton_counts.txt
wc -l ./tmp-dir-list1-no-
singletons/* | sed "s/\.\/.*\///" | sed "s/\.txt.*$//" | sed "s/^\s*//" | sed "s/\s/\t/" >
neg_clean_without_singleton_counts.txt
```

NOTES

Geoffrey Hannigan 12 Jan 2016

Do this singleton removal because sometimes reads can be incorrectly called and this can lead to a samll number of sequences from samples being binned in the negative control samples.

Negative Control Removal- Virome

Step 31.

Get singleton list for reference.

cmd COMMAND

```
mkdir ./tmp-no-singleton-list
for file in $(ls ./tmp-dir-list1-no-singletons); do
    cut -f 2 ./tmp-dir-list1-no-singletons/$file > ./tmp-no-singleton-list/${file}
done
```

Negative Control Removal- Virome

Step 32.

For the purposes of perl filtering, it will be easiest to add @@1 to the end of each sequence ID, using sed.

cmd COMMAND

```
mkdir ./tmp-dir-gi-contigs-perl-filter-format
for file in $(ls ./tmp-dir-gi-contigs); do
    sed '/\@\@/!s/$/\@\@1/' ./tmp-dir-gi-contigs/$file | sed '/>/!s/@@1//' > ./tmp-dir-gi-
contigs-perl-filter-format/${file}
done
```

Negative Control Removal- Virome

Step 33.

Use grep to get all of the sequence names that contain.

cmd COMMAND

```
echo Neg Cleaning - Removing negative sequences from fasta files using perl script...
neg.contig.clean () {
    # INPUT1 = Patient number to use (ie "1")
    # $27 = SiteID
    # $30 = Time Point
    # $1 = Sample ID
    export PAT=$(awk --assign num=$1 --
assign time=$2 '$27 == num && $30 == time { print $1 }' Healthy_human_virome1_mapping.tsv)
```

Negative Control Removal- Virome

Step 34.

Also set which file name is the negative control sample.

cmd COMMAND

```
export NEG_PAT=$(awk --assign num=$1 --
assign time=$2 '$27 == num && $29 == "Neg" && $30 == time { print $1 }' Healthy_human_virom
e1_mapping.tsv)
```

NOTES

Geoffrey Hannigan 12 Jan 2016

Note that in this case an error will return because there is no negative for patient 1 included.

Negative Control Removal- Virome

Step 35.

Make a directory for the specific patient info.

cmd COMMAND

```
mkdir ./patients/patient_${1}_${2}
```

Negative Control Removal- Virome

Step 36.

Copy the patient files to the patient${1} directory.

cmd COMMAND

```
echo $PAT | sed 's/\ /\n/g' | xargs -I {} --max-procs=1 sh -c "cp ./tmp-dir-gi-contigs-perl-filter-format/{}* ./patients/patient_${1}_${2}"
```

The sed command is needed to parse out the words found in $PAT which will be separated by spaces

Negative Control Removal- Virome

Step 37.

Remove all of the fasta sequences that have GI numbers matching their corresponding negative control list of GI numbers.

LINK: https://figshare.com/articles/The_Human_Skin_dsDNA_Virome_Topographical_and_Temporal_Diversity_Genetic_Enrichment_and_Dynamic_Associations_with_the_Host_Microbiome/1281248

cmd COMMAND

```
ls ./patients/patient_${1}_${2}/ | xargs -I {} --max-procs=15 sh -c "perl /project/egricelab/bin/filter_fasta_file.pl -v -l ./tmp-no-singleton-list/${NEG_PAT}* -i ./patients/patient_${1}_${2}/{} -o ./tmp-neg-clean/{} -id_regex '.*@@(\d+)'"
    wait
}
```

NOTES

Geoffrey Hannigan 12 Jan 2016

The perl script filter_fasta_file.pl was used above for filtering fasta sequences based on their sequence ID (in the case above, the gi number ID) and can be found in the supplementary information.

Negative Control Removal- Virome

Step 38.

Export the above subroutine so that it can be used in xargs for faster running on our server.

cmd COMMAND

```
export -f neg.contig.clean
```

Negative Control Removal- Virome

Step 39.

Make patients dif for the above function's output.

cmd COMMAND

```
mkdir ./patients
```

Negative Control Removal- Virome

Step 40.

Use a loop of 20 numbers to cover all of the possible patient numbers in the function neg.contig.clean. Some of these will return ERRORS since not all 20 patients are included in the analyses, but this will push through the errors.

cmd COMMAND

```
seq 1 20 | xargs -I {} --max-procs=20 sh -c 'bsub -n 16 -K -q max_mem64 neg.contig.clean {} 1' & seq 1 20 | xargs -I {} --max-procs=20 sh -c 'bsub -n 16 -K -q max_mem64 neg.contig.clean {} 2' & seq 1 20 | xargs -I {} --max-procs=20 sh -c 'bsub -n 16 -K -q max_mem64 neg.contig.clean {} 3'
wait
```

Negative Control Removal- Virome

Step 41.

Return the fasta files back to their standard "no-@-symbol" format.

cmd COMMAND

```
mkdir ./negative_clean_seqs
for file in $(ls ./tmp-neg-clean); do
    sed 's/@@.*$//' ./tmp-neg-clean/$file | sed 's/@/ /g' > ./negative_clean_seqs/${file}
done
```

Negative Control Removal- Virome

Step 42.

Clean up the negative control cleaned file names.

cmd COMMAND

```
for file in $(ls ./negative_clean_seqs); do
    mv "./negative_clean_seqs/${file}" "./negative_clean_seqs/${file/%.txt*/.fa}"
done
```

Negative Control Removal- Virome

Step 43.

Visualize the results to see how many sequences were removed and how the virus/phage annotation changed. Compare the number of sequences before and after negative control cleaning.

cmd COMMAND

```
#Before
wc -l ./patients/*/* | grep -
v total | sed s'/^\s*//' | sed s'/\s/\t/' | sed s'/\.\/.*\///g' | sed s'/_R1.*//' | awk '{p
rintf $2"\t"$1"\t""before""\n"}' | sort | sed '1 s/^/neg-clean-seqs\tneg-clean-
names\tBefore_or_after\n/' > sequence_counts_before_neg_clean.txt
#After
wc -l ./negative_clean_seqs/* | grep -
v total | sed s'/^\s*//' | sed s'/\s/\t/' | sed s'/\.\/.*\///g' | sed s'/_R1.*//' | awk '{p
rintf $2"\t"$1/2"\t""after""\n"}' | sort > sequence_counts_after_neg_clean.txt
```

Negative Control Removal- Virome

Step 44.

Merge the files for graphing in R

cmd COMMAND

```
cat sequence_counts_before_neg_clean.txt sequence_counts_after_neg_clean.txt > sequence_cou
nts_neg_clean_for_R.txt
```

Negative Control Removal- Virome

Step 45.

Use R to graph the neg clean summary info and get the summary stats.

cmd COMMAND

```
mkdir ./neg_cntrl_clean_summary_stats
Rscript neg_clean_count_before_after_graphing.R "./sequence_counts_neg_clean_for_R.txt" "./
neg_cntrl_clean_summary_stats"
```

Pre-Processing Results

Step 46.

We then wanted to see how many low-quality and contaminated sequences we removed from our raw sequences. In order to do so, we had to determine how many sequences were in each file.

cmd COMMAND

```
mkdir ./sequence_count_stats
```

NOTES

Geoffrey Hannigan 12 Jan 2016

For simplicity, we are only looking at the R1 reads.

Pre-Processing Results

Step 47.

Count total number of raw sequences (and median sequence lengths).

LINK:

https://figshare.com/articles/The_Human_Skin_dsDNA_Virome_Topographical_and_Temporal_Diversity_Genetic_Enrichment_and_Dynamic_Associations_with_the_Host_Microbiome/1281248

cmd COMMAND

```
for file in ./raw_fastq/*_R1.fastq; do
    name=`echo "$file" | sed 's/\.\/raw_fastq\///g' | sed 's/_R1\.fastq//g'`
    count=`cat $file | wc -l | awk '{print $1/4}'`
    echo -e "$name\t$count" >> ./sequence_count_stats/raw_sequence_counts.tsv
    perl calculate_fastq_median_length.pl $file >> ./sequence_count_stats/raw_median_seq_le
ngth.tsv
done
```

NOTES

Geoffrey Hannigan 12 Jan 2016

The perl script calculate_fastq_median_length.pl is used in this step and can be obtained from the supplementary information.

Pre-Processing Results

Step 48.

Count adapter trimmed (cutadapt) and quality trimmed (fastx) sequences (and median sequence lengths).

cmd COMMAND

```
for file in ./trimmed_fastq/*_R1.fastq; do
    name=`echo "$file" | sed 's/\.\/trimmed_fastq\///g' | sed 's/_R1\.fastq//g'`
    count=`cat $file | wc -l | awk '{print $1/4}'`
    echo -e "$name\t$count" >> ./sequence_count_stats/trimmed_sequence_counts.tsv
    perl calculate_fastq_median_length.pl $file >> ./sequence_count_stats/trimmed_median_se
q_length.tsv
done
```

Pre-Processing Results

Step 49.

Count human decontaminated sequences (and median sequences lengths).

cmd COMMAND

```
for file in ./clean_fastq/*_R1.fastq; do
    name=`echo "$file" | sed 's/\.\/clean_fastq\///g' | sed 's/_R1\.fastq//g'`
    count=`cat $file | wc -l | awk '{print $1/4}'`
    echo -e "$name\t$count" >> ./sequence_count_stats/human_deconseq_sequence_counts.tsv
    perl calculate_fastq_median_length.pl $file >> ./sequence_count_stats/human_cleaned_med
ian_seq_length.tsv
done
```

Pre-Processing Results

Step 50.

Count PhiX decontaminated sequences (and median sequence lengths) (whole metagenome only).

cmd COMMAND

```
for file in ./clean_phix_fastq/*_R1.fastq; do
    name=`echo "$file" | sed 's/\.\/clean_phix_fastq\///g' | sed 's/_R1\.fastq//g'`
    count=`cat $file | wc -l | awk '{print $1/4}'`
    echo -e "$name\t$count" >> ./sequence_count_stats/phix_clean_sequence_counts.tsv
    perl calculate_fastq_median_length.pl $file >> ./sequence_count_stats/clean_phix_median
_seq_length.tsv
done
```

Pre-Processing Results

Step 51.

Count Negative Control decontaminated sequences (and median sequence lengths) (virome only).

LINK:

https://figshare.com/articles/The_Human_Skin_dsDNA_Virome_Topographical_and_Temporal_Divers

ity_Genetic_Enrichment_and_Dynamic_Associations_with_the_Host_Microbiome/1281248

cmd COMMAND

```
for file in ./negative_clean_seqs/*_R1.fa; do
    name=`echo "$file" | sed 's/\.\/negative_clean_seqs\///g' | sed 's/_R1\.fa//g'`
    count=`cat $file | wc -l | awk '{print $1/2}'`
    echo -e "$name\t$count" >> ./sequence_count_stats/negative_clean_sequence_counts.tsv
    perl calculate_fasta_median_length.pl $file >> ./sequence_count_stats/negative_clean_se
quence_length_medians.tsv
done
```

NOTES

Geoffrey Hannigan 12 Jan 2016

The perl script calculate_fasta_median_length.pl is used in this step and is obtainable from the supplementary information.

Pre-Processing Results

Step 52.

We further validated our database quality by comparing the levels of 16S rRNA sequences between the virome and whole metagenome, as well as comparing the number of reads matching from the virome to the whole metagenome, and vice versa. First we quantified the levels of 16S rRNA matching sequences.

cmd COMMAND

```
mkdir ./estimated_contamination_ribosome_hits
mkdir ./estimated_contamination_ribosome_hits/whole_metagenome_subsampled_10k
mkdir ./estimated_contamination_ribosome_hits/whole_metagenome_subsampled_10k_fasta
mkdir ./estimated_contamination_ribosome_hits/virome_subsampled_10k
```

Make dir for output

Pre-Processing Results

Step 53.

Run blastn to attempt to quantify level of contamination using gg16SrRNA database.

Pre-Processing Results

Step 54.

Subsample files.

cmd COMMAND

```
for file in $(ls /project/egricelab/SkinMetVirome_tmp_files/clean_phix_fastq/*_R1* | sed 's
/.*\///g'); do
    NAME=$(echo ${file} | sed 's/\.fastq//')
    echo Subsampling ${NAME}...
    /project/egricelab/meiselj_software/idba_ud-1.0.9/bin/fq2fa /project/egricelab/SkinMetV
irome_tmp_files/clean_phix_fastq/${NAME}.fastq ./estimated_contamination_ribosome_hits/whol
e_metagenome_subsampled_10k_fasta/${NAME}.fasta
    /project/egricelab/meiselj_software/seqtk-
master/seqtk sample ./estimated_contamination_ribosome_hits/whole_metagenome_subsampled_10k
_fasta/${NAME}.fasta 10000 > ./estimated_contamination_ribosome_hits/whole_metagenome_subsa
mpled_10k/${NAME}.fasta
done
for file in $(ls ./negative_clean_seqs); do
    echo Subsampling ${file}...
    /project/egricelab/meiselj_software/seqtk-
master/seqtk sample ./negative_clean_seqs/$file 10000 > ./estimated_contamination_ribosome_
hits/virome_subsampled_10k/${file}
done
```

Pre-Processing Results

Step 55.

Move the blastx output to its own directory.

cmd COMMAND

```
echo Performing blastx for virome...
mkdir ./estimated_contamination_ribosome_hits/green_genes_blastx_hits

for file in $(ls ./estimated_contamination_ribosome_hits/virome_subsampled_10k); do
    blastn -query ./estimated_contamination_ribosome_hits/virome_subsampled_10k/${file} -
out ./estimated_contamination_ribosome_hits/green_genes_blastx_hits/${file} -
db /project/egricelab/ghanni_software/ribosome_ref_test/green_genes_for_virus_db -
outfmt 6 -num_threads 2 -max_target_seqs 1 -evalue 1e-5
done
```

Pre-Processing Results

Step 56.

Get the numbers of hits for the virome samples. First get the blastx hits for the virome against the greengenes database.

cmd COMMAND

```
wc -
l ./estimated_contamination_ribosome_hits/green_genes_blastx_hits/* | sed 's/\.\/estimated_
contamination_ribosome_hits\/green_genes_blastx_hits\///' | sed 's/_R1\.fa//' | grep -
v total | sed 's/^\s*//' | sed 's/\s/\t/g' | sed 's/$/\tVirome_16S_hits/' > ./estimated_con
tamination_ribosome_hits/virome_blastx_hits.tsv
```

Pre-Processing Results

Step 57.

Get the number of sequences in each file as well.

cmd COMMAND

```
wc -
l ./estimated_contamination_ribosome_hits/virome_subsampled_10k/* | sed 's/\.\/HUMAnN_outpu
t\/neg_clean_fasta_subsampled_10k\///' | sed 's/_R1\.fa//' | grep -
v total | sed 's/^\s*//' | sed 's/\s/\t/g' | awk '{ print $1/2"\t"$2 }' > ./estimated_conta
mination_ribosome_hits/virome_subsampled_numbers.tsv
```

Pre-Processing Results

Step 58.

Paste together the files.

cmd COMMAND

```
paste ./estimated_contamination_ribosome_hits/virome_blastx_hits.tsv ./estimated_contaminat
ion_ribosome_hits/virome_subsampled_numbers.tsv > ./estimated_contamination_ribosome_hits/v
irome_contamination_info.tsv
```

Pre-Processing Results

Step 59.

Run the same blast on the whole metagenome samples for a comparison.

cmd COMMAND

```
echo Estimating bacterial levels in whole metagenome using blastx against the gg16SrRNA dat
abase...
mkdir ./estimated_contamination_ribosome_hits/green_genes_blastx_hits_whole_metagenome
for file in $(ls ./estimated_contamination_ribosome_hits/whole_metagenome_subsampled_10k);
do
    blastn -
query ./estimated_contamination_ribosome_hits/whole_metagenome_subsampled_10k/${file} -
out ./estimated_contamination_ribosome_hits/green_genes_blastx_hits_whole_metagenome/${file
} -db /project/egricelab/ghanni_software/ribosome_ref_test/green_genes_for_virus_db -
outfmt 6 -num_threads 2 -max_target_seqs 1 -evalue 1e-5
done
```

Pre-Processing Results

Step 60.

Format these numbers into a table for R. First get together a list of the blastx hits against green

genes.

cmd COMMAND

```
wc -
l ./estimated_contamination_ribosome_hits/green_genes_blastx_hits_whole_metagenome/* | sed
's/\.\/estimated_contamination_ribosome_hits\/green_genes_blastx_hits_whole_metagenome\///g
' | sed 's/_R1_subsampled\.fa//g' | grep -
v total | sed 's/^\s*//' | sed 's/\s/\t/g' | sed 's/$/\tWhole_Metagenome_16S_hits/' > ./est
imated_contamination_ribosome_hits/whole_metagenome_blastx_hits.tsv
```

Pre-Processing Results

Step 61.

Make list of the total number of sequences.

cmd COMMAND

```
wc -
l ./estimated_contamination_ribosome_hits/whole_metagenome_subsampled_10k/* | sed 's/\/proj
ect\/egricelab\/metagenome_subsampled_5000\///' | sed 's/_R1_subsampled\.fa//' | grep -
v total | sed 's/^\s*//' | sed 's/\s/\t/g' | awk '{ print $1/2"\t"$2 }' > ./estimated_conta
mination_ribosome_hits/whole_metagenome_subsampled_numbers.tsv
```

Pre-Processing Results

Step 62.

Paste together these files.

cmd COMMAND

```
paste ./estimated_contamination_ribosome_hits/whole_metagenome_blastx_hits.tsv ./estimated_
contamination_ribosome_hits/whole_metagenome_subsampled_numbers.tsv > ./estimated_contamina
tion_ribosome_hits/whole_metagenome_contamination_info.tsv
```

Pre-Processing Results

Step 63.

Cat together the files for input into R.

cmd COMMAND

```
cat ./estimated_contamination_ribosome_hits/virome_contamination_info.tsv ./estimated_conta
mination_ribosome_hits/whole_metagenome_contamination_info.tsv > ./estimated_contamination_
ribosome_hits/total_contamination_info_for_R.tsv
```

Pre-Processing Results

Step 64.

We also quantified the number of reads mapping from one overall dataset to the other corresponding dataset (virome to whole metagenome and vice versa).

cmd COMMAND

```
mkdir ./virome_whole_metagenome_blast
mkdir ./virome_whole_metagenome_blast/blastdb_whole_metagenome
mkdir ./virome_whole_metagenome_blast/blastn_virome_results
mkdir ./virome_whole_metagenome_blast/blastdb_virome
mkdir ./virome_whole_metagenome_blast/blastn_whole_metagenome_results
mkdir ./virome_whole_metagenome_blast/whole_metagenome_subsampled_50k
mkdir ./virome_whole_metagenome_blast/virome_subsampled_50k
```

Make directories for output.

Pre-Processing Results

Step 65.

Subsample files.

cmd COMMAND

```
for file in $(ls ./clean_phix_fastq/*_R1* | sed 's/.*\///g'); do
    NAME=$(echo ${file} | sed 's/\.fastq//')
    echo Subsampling ${NAME}...
    idba_ud-1.0.9/bin/fq2fa ./clean_phix_fastq/${NAME}.fastq ./estimated_contamination_ribo
some_hits/whole_metagenome_subsampled_10k_fasta/${NAME}.fasta
```

```
    seqtk sample ./estimated_contamination_ribosome_hits/whole_metagenome_subsampled_10k_fa
sta/${NAME}.fasta 50000 > ./virome_whole_metagenome_blast/whole_metagenome_subsampled_50k/$
{NAME}.fasta
done
for file in $(ls ./negative_clean_seqs); do
    echo Subsampling ${file}...
    seqtk sample ./negative_clean_seqs/${file} 50000 > ./virome_whole_metagenome_blast/viro
me_subsampled_50k/${file}
done
```

Pre-Processing Results

Step 66.

Generate blastn reference from whole metagenome samples and perform blastn of virome reads against database.

cmd COMMAND

```
run_blastn_virome_to_wm_and_back () {
    if [ -f /etc/profile.d/modules.sh ]; then
        source /etc/profile.d/modules.sh
    fi
    module load ncbi-blast-2.2.0
    export NAME=$(echo $1 | sed 's/_R1\.fasta//g')
    echo Sample ID is ${NAME}
    makeblastdb -dbtype nucl -
in ./virome_whole_metagenome_blast/whole_metagenome_subsampled_50k/$1 -
out ./virome_whole_metagenome_blast/blastdb_whole_metagenome/${NAME}_db
```

Pre-Processing Results

Step 67.

Set the corresponding virus sample name to a new variable.

cmd COMMAND

```
export VIRUSID=$(awk --
assign name=${NAME} '$5 == name { print $7 }' ./SkinMet_and_Virome_001_metadata.tsv)
    echo Virus Sample ID for ${NAME} is ${VIRUSID}
    blastn -query ./virome_whole_metagenome_blast/virome_subsampled_50k/${VIRUSID}_R1.fa -
out ./virome_whole_metagenome_blast/blastn_virome_results/${VIRUSID}.txt -
db ./virome_whole_metagenome_blast/blastdb_whole_metagenome/${NAME}_db -outfmt 6 -
max_target_seqs 1 -evalue 1e-5
```

Pre-Processing Results

Step 68.

Perform the opposite blastn as well, first making a new database.

cmd COMMAND

```
makeblastdb -dbtype nucl -
in ./virome_whole_metagenome_blast/virome_subsampled_50k/${VIRUSID}_R1.fa -
out ./virome_whole_metagenome_blast/blastdb_virome/${VIRUSID}_db
    blastn -query ./virome_whole_metagenome_blast/whole_metagenome_subsampled_50k/$1 -
out ./virome_whole_metagenome_blast/blastn_whole_metagenome_results/${NAME}.txt -
db ./virome_whole_metagenome_blast/blastdb_virome/${VIRUSID}_db -outfmt 6 -
max_target_seqs 1 -evalue 1e-5
}
export -f run_blastn_virome_to_wm_and_back
mkdir ./virome_whole_metagenome_blast/error-out
ls ./virome_whole_metagenome_blast/whole_metagenome_subsampled_50k | xargs -I {} --max-
procs=328 sh -c "run_blastn_virome_to_wm_and_back {}"
```

Pre-Processing Results

Step 69.

Generate a table of the blastn hit abundances for each sample, split by whether it is virome or whole metagenome.

cmd COMMAND

```
wc -
l ./virome_whole_metagenome_blast/blastn_virome_results/* | sed 's/^ *//' | sed 's/ /\t/' |
 sed 's/\.\/.*\(MG\)/\1/' | sed 's/\.txt//' | grep -
v total > ./virome_whole_metagenome_blast/virome_to_metagenome_blastn_results.tsv
wc -
l ./virome_whole_metagenome_blast/blastn_whole_metagenome_results/* | sed 's/^ *//' | sed '
s/ /\t/' | sed 's/\.\/.*\(MG\)/\1/' | sed 's/\.txt//' | grep -
v total > ./virome_whole_metagenome_blast/metagenome_to_virome_blastn_results.tsv
wc -
l ./virome_whole_metagenome_blast/virome_subsampled_50k/* | sed 's/^ *//' | sed 's/ /\t/' |
 sed 's/\.\/.*\(MG\)/\1/' | sed 's/_R1\.fa//' | grep -
v total > ./virome_whole_metagenome_blast/virome_to_metagenome_sequence_counts.tsv
wc -
l ./virome_whole_metagenome_blast/whole_metagenome_subsampled_50k/* | sed 's/^ *//' | sed '
s/ /\t/' | sed 's/\.\/.*\(MG\)/\1/' | sed 's/_R1\.fasta//' | grep -
v total > ./virome_whole_metagenome_blast/metagenome_to_virome_sequence_counts.tsv
```

Pre-Processing Results

Step 70.

Combine the sequence count and hit tables.

cmd COMMAND

```
awk 'FNR==NR { a[$2]=$1; next } $2 in a { print $2"\t"$1"\t"a[$2] }' ./virome_whole_metagen
ome_blast/virome_to_metagenome_blastn_results.tsv ./virome_whole_metagenome_blast/virome_to
_metagenome_sequence_counts.tsv > ./virome_whole_metagenome_blast/virome_to_metagenome_shar
ed_results.tsv
awk 'FNR==NR { a[$2]=$1; next } $2 in a { print $2"\t"$1"\t"a[$2] }' ./virome_whole_metagen
ome_blast/metagenome_to_virome_blastn_results.tsv ./virome_whole_metagenome_blast/metagenom
e_to_virome_sequence_counts.tsv > ./virome_whole_metagenome_blast/metagenome_to_virome_shar
ed_results.tsv
```

NOTES

Geoffrey Hannigan 12 Jan 2016

Some of the whole metagenome samples did not have corresponding virome samples and so just had zero hits recorded. This will be cleaned up downstream in R.