

Jester King Data

For taxonomic assignment, you can use the FASTQ files which are in
`/vol_c/minION_run_data/JesterKing-20170606.fastq`

For QC use the FAST5 files are in
`/vol_c/minION_run_data/20170606_0419_JesterKing2/workspace/` in
subdirectories like 0, 1, 2, etc.

Kraken databases

The databases for use in Kraken are located in:

`/vol_c/public_data/kraken_dbs/`

Kraken commands:

Against minikraken db:

```
kraken --db /vol_c/public_data/kraken_dbs/minikraken_20141208/ \  
      --threads 10 \  
      --fastq-input \  
      --output jesterking.kraken \  
      --preload \  
      /vol_c/minION_run_data/JesterKing-20170606.fastq
```

```
kraken-report --db \  
/vol_c/public_data/kraken_dbs/minikraken_20141208/  
jesterking.kraken \  
> jesterking.kraken.report
```

Against yeast db:

```
kraken --db /vol_c/public_data/kraken_dbs/yeast/ \  
      --threads 10 \  
      --fastq-input \  
      --output jesterking.yeast.kraken \  
      --preload \  
      /vol_c/minION_run_data/JesterKing-20170606.fastq
```

```
kraken-report --db \  
/vol_c/public_data/kraken_dbs/yeast/ jesterking.yeast.kraken \  
> jesterking.yeast.kraken.report
```

Centrifuge databases

The databases for use with Centrifuge are located in:

```
/vol_c/public_data/centrifuge_dbs/
```

Centrifuge commands eg:

Default p_compressed database:

```
centrifuge -x /vol_c/public_data/centrifuge_dbs/p_compressed \  
-U /vol_c/minION_run_data/JesterKing-20170606.fastq \  
-S jk.centrifuge.class\  
--report-file jk.centrifuge.report \  
-p 10
```

```
# generate kraken-style report
```

```
centrifuge-kreport -x
```

```
/vol_c/public_data/centrifuge_dbs/p_compressed jk.centrifuge.class
```

```
> jk.centrifuge.kreport
```

```
# species level report above 1%
```

```
cat jk.centrifuge.kreport | awk '$4=="S"' | awk '$1>=1'
```

```
# to consume the default centrifuge report
```

```
# use this obvious one-liner
```

```
(head -n 1 jk.centrifuge.report && tail -n +3 jk.centrifuge.report  
| sort -nr -t$'\t' -k 5) | more
```

Search whole of nt

```
centrifuge -x /vol_c/public_data/centrifuge_dbs/nt \  
-U /vol_c/minION_run_data/JesterKing-20170606.fastq \  
-S jk.centrifuge.nt.class \  
--report-file jk.centrifuge.nt.report \  
-p 10
```

```
# generate kraken-style report
```

```
centrifuge-kreport -x /vol_c/public_data/centrifuge_dbs/nt
```

```
jk.centrifuge.nt.class > jk.centrifuge.nt.kreport &
```

```
# species level report above 1%
```

```
cat jk.centrifuge.nt.kreport | awk '$4=="S"' | awk '$1>=1'
```

```
# to consume the default centrifuge report
```

```
# use this obvious one-liner
```

```
(head -n 1 jk.centrifuge.nt.report && tail -n +3  
jk.centrifuge.nt.report | sort -nr -t$'\t' -k 5) | more
```

Classification summary

See sheet

https://docs.google.com/spreadsheets/d/1IAAt3gt78pmmu4jtWM0B6KdBY3C_ncPC3o943fKA_Neo4/edit?usp=sharing

Removing new lines

```
grep -v -e '^$' foo.txt > bar.txt
```

Miniasm + minimap

```
# all against all
minimap -Sw5 -L100 -m0 \
    -t8 /vol_c/minION_run_data/JesterKing-20170606.fastq \
        /vol_c/minION_run_data/JesterKing-20170606.fastq | \
    gzip -1 > jk.paf.gz
```

```
miniasm -f /vol_c/minION_run_data/JesterKing-20170606.fastq
jk.paf.gz > jk.gfa
```

```
awk '/^S/{print ">"$2"\n"$3}' jk.gfa | fold > jk.assembly.fa
```

```
# blast
blastn -query jk.assembly.fa -db
/vol_c/public_data/blast_databases/nt -outfmt 6 -num_threads 10
-max_target_seqs 1 | more
```

Top ten reads by length

```
(head -n 1 /vol_c/minION_run_data/20170606_0419_JesterKing2/sequencing_summary.txt
&& tail -n +3
/vol_c/minION_run_data/20170606_0419_JesterKing2/sequencing_summary.txt | sort -nr
-k12) | awk '{print $2"\t"$12}' | head
```

```
cat /vol_c/minION_run_data/JesterKing-20170606.fastq | paste - - - - | grep  
a1d86fb7-67cd-411a-bec1-b311b60be806 | awk -F"\t" '{print ">"$1"\n"$2}' > longest.fa
```

Create subsets of fastq by length

```
cat /vol_c/minION_run_data/JesterKing-20170606.fastq | paste - - - - | awk -F"\t" 'length($2)  
> 1000' | awk -F"\t" '{print $1"\n"$2"\n"$3"\n"$4}' > jk.1000.fq &
```

```
cat /vol_c/minION_run_data/JesterKing-20170606.fastq | paste - - - - | awk -F"\t" 'length($2)  
> 10000' | awk -F"\t" '{print $1"\n"$2"\n"$3"\n"$4}' > jk.10000.fq &
```

```
cat /vol_c/minION_run_data/JesterKing-20170606.fastq | paste - - - - | awk -F"\t" 'length($2)  
> 5000' | awk -F"\t" '{print $1"\n"$2"\n"$3"\n"$4}' > jk.5000.fq &
```

Online taxonomic classification

www.onecodex.com

Download Lactobacillus backii genomes

```
ncbi-genome-download -F fasta --genus "Lactobacillus backii" bacteria
```

Plasmid discovery

Run blastn:

```
blastn -query jk.assembly.fa -db /vol_c/public_data/blast_databases/nt -num_threads 10  
-eval 1e-10 -outfmt '6 qseqid qlen sseqid stitle slen evlaue length pident' >  
jk.assembly.nt.blast &
```

Look where the query length is within 10% of the length of the hit, and where the description contains the words “plasmid” and “complete”

```
grep plasmid jk.assembly.nt.blast | grep complete | awk -F"\t" '$2/$5>=0.9' | awk -F"\t"  
'$2/$5<=1.1' | more
```