Analysis of unmapped P. major reads

Aims

Subsetting of bam files

Made subsets of bam files for testing using subset_bams.sh:

```bash
#!/bin/bash

conda activate samtools

mkdir subsets

for file in *.bam
do
    FILENAME="$file"
    FILENAME=${FILENAME%.bam*}
    echo $FILENAME
    samtools view -bo "$FILENAME"_subset.bam -s 123.001 "$file"
done

conda deactivate

# move subset-bams into the subset directory
mv *_subset.bam subsets/
```

Setup miniconda environments

Look at file sizes and raw data more generally

Count number of reads in bam files

I counted the number of reads in the bam files with the shell-script `num_reads_bams.sh`:

```bash
#!/bin/bash


touch num_reads.txt

for file in *.bam
do
    printf '%s\t%s\n' $file $(samtools view -@ 8 -c $file)
```

```
done > num_reads.txt
```

Distribution of read lengths

Counts of read lengths can be calculated with the following shell command.

```
# get counts (2. column) of read of different length (1. column)
samtools stats S1_EKDN230004350-1A_HNW2NDSX_sorted_dedup_unmapped_subset.bam | grep ^RL | cut -f 2-
```

I wrote a Python-script to plot a histogram of the read length of sample bam file:

```
# Plot histograms of read length distribution of sample bam files


import subprocess

import matplotlib.pyplot as plt

shell_command = 'ls | grep .bam'

OUTPUT_STREAM = subprocess.run(
    shell_command, capture_output=True, shell=True, text=True, check=True)

bam_files = OUTPUT_STREAM.stdout.split('\n')[:-1]

#
fig, axs = plt.subplots(4, 5, sharey=True, sharex=True)


positions = {0: (0, 0),
             1: (0, 1),
             2: (0, 2),
             3: (0, 3),
             4: (0, 4),
             5: (1, 0),
             6: (1, 1),
             7: (1, 2),
             8: (1, 3),
             9: (1, 4),
             10: (2, 0),
             11: (2, 1),
             12: (2, 2),
             13: (2, 3),
             14: (2, 4),
             15: (3, 0),
             16: (3, 1),
             17: (3, 2),
             18: (3, 3),
               19: (3, 4)}


for i, file in enumerate(bam_files):

    sample = file.split('_')[0]
    samtools_command = f'samtools stats {file} | grep ^RL | cut -f 2-
'.format(
        file)

    OUTPUT_STREAM = subprocess.run(
        samtools_command, capture_output=True, shell=True, text=True,
check=True)

    rows = OUTPUT_STREAM.stdout
    split_rows = rows.split('\n')
    split_split_rows = [row.split('\t') for row in split_rows][:-1]
    read_lens = ([int(entry[0]) for entry in split_split_rows])
    read_lens.append(151)
    read_counts = ([int(entry[1]) for entry in split_split_rows])
    axs[positions[i]].stairs(read_counts, read_lens, fill=True)
    axs[positions[i]].set_title(sample, pad=3.0)
    plt.yscale('log')

fig.text(0.5, 0.04, 'Read length', ha='center', va='center', fontsize=18)
fig.text(0.06, 0.5, 'Read counts', ha='center',
         va='center', rotation='vertical', fontsize=18)
plt.show()

print('done')
```

Kraken2 analysis

```


## Download additional genomes



# Parus major
ncbi-genome-download --section genbank vertebrate_other -A GCA_001522545.3
-F fasta,assembly-report -p 4 -r 3 -o /work/mnikvell/data/genomes/
gzip -d
/work/data/genomes/genbank/vertebrate_other/GCA_001522545.3/GCA_001522545.3
_Parus_major1.1_genomic.fna.gz

# Gallus gallus
ncbi-genome-download --section genbank vertebrate_other -A GCA_016699485.1
-F fasta,assembly-report -p 4 -r 3 -o /work/mnikvell/data/genomes/
gzip -d
/work/data/genomes/genbank/vertebrate_other/GCA_016699485.1/GCA_016699485.1
_bGalGal1.mat.broiler.GRCg7b_genomic.fna.gz
```

```
# Haemoproteus tartakovskyi
ncbi-genome-download --section genbank invertebrate -A GCA_001625125.1 -F
fasta,assembly-report -p 4 -r 3 -o /work/mnikvell/data/genomes
gzip -d
/work/mnikvell/data/genomes/genbank/protozoa/GCA_001625125.1/GCA_001625125.
1_ASM162512v1_genomic.fna.gz
```

Shell script to install libraries and add genomes

TODO

```
#!/bin/bash


echo 'taxonomy'
kraken2-build --download-taxonomy --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/

echo 'archaea'
kraken2-build --download-library archaea --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/

echo 'bacteria'
kraken2-build --download-library bacteria --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/

echo 'plasmid'
kraken2-build --download-library plasmid --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/

echo 'viral'
kraken2-build --download-library viral --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/

echo 'human'
kraken2-build --download-library human --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/

echo 'fungi'
kraken2-build --download-library fungi --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/

echo 'plant'
kraken2-build --download-library plant --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/

echo 'protozoa'
kraken2-build --download-library protozoa --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/

echo 'UniVec_Core'
```

```
kraken2-build --download-library UniVec_Core --threads 3 --db
/work/mnikvell/data/Kraken2/dbs/full/
```

Shell script to build database on lido3-cluster

```
#!/bin/bash -l
#SBATCH --partition=long
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --time=2-00:00:00
#SBATCH --cpus-per-task=32
#SBATCH --mem-per-cpu=6G
#SBATCH --job-name=kraken_build_job
#SBATCH --mail-user=nikolas.vellnow@tu-dortmund.de
#SBATCH --mail-type=All

conda activate kraken

DB_NAME=full
DB_PATH=/scratch/mnikvell/kraken_job_${SLURM_JOBID}/${DB_NAME}/
OUT_PATH=/work/mnikvell/data/Kraken2/dbs/

echo "db name: ${DB_NAME}"
echo "db path: ${DB_PATH}"
echo "output path: ${OUT_PATH}"

# create directories in scratch-dir
rm -rf /scratch/mnikvell/kraken_job_${SLURM_JOBID}/
mkdir -p /scratch/mnikvell/kraken_job_${SLURM_JOBID}/
mkdir -p /scratch/mnikvell/kraken_job_${SLURM_JOBID}/${DB_NAME}


# copy db data
cp -a /work/mnikvell/data/Kraken2/dbs/full_data/. $DB_PATH

# scratch directory
echo "content of scratch dir: $(ls -R /scratch/mnikvell/)"

# move to job directory
cd /scratch/mnikvell/kraken_job_${SLURM_JOBID}/

# add genomes (already downloaded) to library

echo 'genome great tit'
kraken2-build --add-to-library
/work/mnikvell/data/genomes/genbank/vertebrate_other/GCA_001522545.3/GCA_00
1522545.3_Parus_major1.1_genomic.fna \
--db "${DB_PATH}" --threads 32

echo 'genome chicken'
kraken2-build --add-to-library
/work/mnikvell/data/genomes/genbank/vertebrate_other/GCA_016699485.1/GCA_01
6699485.1_bGalGal1.mat.broiler.GRCg7b_genomic.fna \
--db "${DB_PATH}" --threads 32

echo 'genome blood parasite'
kraken2-build --add-to-library
/work/mnikvell/data/genomes/genbank/protozoa/GCA_001625125.1/GCA_001625125.
1_ASM162512v1_genomic.fna \
--db "${DB_PATH}" --threads 32

# build database
kraken2-build --build --db "${DB_PATH}" --threads 32

# clean unnecessary files
kraken2-build --clean --db "${DB_PATH}" --threads 32

# copy outputs back to
cp -a $DB_PATH $OUT_PATH

rm -rf /scratch/mnikvell/kraken_job_${SLURM_JOBID}/

conda deactivate
```

Shell script to build database

TODO

```
kraken2 --db /work/mnikvell/data/Kraken2/dbs/EuPathDB46 --output
/work/mnikvell/data/Kraken2/outputs/output_EuPathDB46_S1 --use-names --
report /work/mnikvell/data/Kraken2/outputs/report_EuPathDB46_S1
/work/mnikvell/data/subsets/S1_EKDN230004350-
1A_HNW2NDSX_sorted_dedup_unmapped_subset.fasta
```