INF-BIO9121/5121 Fall 2017 exam: assembly

The date for this home exam is **November 3rd** (the day after the written exam). Before this day you have to solve the tasks described here, and prepare a 15 to 20-minute presentation to be held on the exam date.

The following contains explanations of the tasks, and directions for what to present. Along the way, there are questions you need to be able to answer to during your presentation. You have around 20 minutes, so don't try to put too much on your slides. I'd rather see some well-presented interesting results, than many slides with too little time for the last ones. Please also present, and explain, the actual command lines for the steps you performed. Do not submit answers in writing.

NOTE that some tasks are for PhD students only!

If you have a pressing question or a problem that stops you from completing the tasks, contact me at karin.lagesen@medisin.uio.no.

Make sure you read the instructions carefully!

Background

You will be using datasets quite similar to the ones we used during the course, but from a different bacterial species: *Meiothermus ruber DSM 1279*. The data was used for a few publications:

- Tindall *et al.*, "Complete genome sequence of Meiothermus ruber type strain (21)" Stand Genomic Sci 3(1): 26-36 (2010). DOI: 10.4056/sigs.1032748
- Illumina Application Note "*De Novo* Assembly of Bacterial Genomes", http://www.illumina.com/content/dam/illumina-marketing/documents/products/appnotes/appnote-nextera-mate-pair-bacteria.pdf
- Chin *et al*, Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data. Nature Methods 10, 563–569 (2013). DOI: doi:10.1038/nmeth.2474

There is no requirement to read these, nor will it help for the exam, but you're obviously welcome to do so.

Please note that the genome of this strain is 3.1Mbp, not 4.6 Mbp!

Data

The input data is located here:

```
/data/exam/assembly/
```

The following files are provided:

- in the `illumina` folder
  - `SRR1800541_R1_50x.fastq` and `SRR1800541_R2_50x.fastq` contain reads from a HiSeq run of the strain subsampled to 50x coverage.
  - `SRR1800541_R1_350x.fastq` and `SRR1800541_R1_350x.fastq` contain reads from a HiSeq run of the strain subsampled to 350x coverage.
  - `ERR76053X_MP_R1_trimmed.fastq` and `ERR76053X_MP_R2_trimmed.fastq` contain the mate pair data from a MiSeq run, trimmed for adaptors and low-quality bases
- in the `pacbio` folder
  - `SRR8118XX_m120803.filtered_subreads.fastq` contains 124x coverage of slightly older PacBio reads (so-called `C2XL` chemistry) in fastq file format (these are the so-called subreads).
- in the `reference` folder
  - `CP005385.1_Meiothermus_ruber_DSM_1279.fna` contains the genome sequence in `fasta` format
  - `CP005385.1_Meiothermus_ruber_DSM_1279.gff` the annotated genes in `gff` format

NOTE there are no MinION data available for this strain.

All tasks

- always use the `assemblathon_stats.pl` script to judge assemblies
- **IMPORTANT** use a minimum gap length of 1 bp for all analysis (not the default from each tool). This affects
  - `assemblathon_stats.pl`
  - `scaffoldgap2bed.py`
- show the commands you used
- for *each 'final' assembly*, show the most relevant metrics.

Part 1: Assembly

Task 1.1: velvet assembly of the Illumina paired end reads only

- first, determine optimal `kmer` size for an assembly with the **paired end reads** only, by creating several assemblies with varying k.
  - use the 50x data for this task
  - do not use the pairing information or `-cov_cutoff` or `-exp_cov` parameters
- second, create a paired end assembly using the k found above
  - Use your obtained optimal `kmer` size and perform a default velvet assembly, this time *using the pairing information in the reads*, with the following settings (and any other you deem necessary)

```
-exp_cov auto
-cov_cutoff auto
```

- call this assembly `velvet_pe`
- **PhD students only** Use the node coverage ipython notebook as we did in the course to analyze the coverage of this assembly.

Questions you need to answer

- show the `kmer` versus N50 plot, argue for which `kmer` you chose
- how much did adding the pairing information help for the assembly quality?
- **PhD students only:** what does velvet report for exp_cov? Show the coverage graph from the notebook. How does this correspond to what you see in the notebook graph?

Task 1.2: velvet assembly of the illumina paired end reads and mate pair reads

- create velvet assembly using PE and MP reads
- use the same `kmer` size as before, add the mate pair reads for a new velvet assembly
- the mate pair reads are already in the orientation that velvet expects them to be (so-called 'forward-reverse')
- include the following settings (and any other you deem necessary)

```
-exp_cov auto
-cov_cutoff auto
```

- call this `velvet_pe+mp`

Questions you need to answer

- what did velvet determine to be the insert size for both libraries, with which standard deviation?
- how do these numbers compare to the ones for the libraries we used during the course?

Task 1.3: SPADES Illumina-only assembly

- create SPADES assembly using a different read set
- use the **350x data** for this task (!)
- perform the SPADES assembly *as described for the course* with the exam data (both paired end and mate pair)
- settings to think about:
  - `-t` number of CPUs to use: do not use more than 3
  - `--memory` memory usage, don't use more than 30
  - `-k` k-mer range: use `21,29,37,43`
  - `--mp1-fr` this tells SPADES the orientation of the mate pairs
- call this assembly `spades_pe+mp`

Questions you need to answer

- which assembly is 'better', the velvet or the SPADES assembly, and why?

Task 1.4: PHD student only! SPADES hybrid assembly

- use the **350x data** for this task (!)
- replace the matepair data with the PacBio reads (the `fastq` file with subreads) as done during the

course
- run the assembly the same as for the previous task
- call this `spades_hybrid`
- NOTE **PHD student only!**

Questions you need to answer

- did replacing the mate-pair data with pacbio reads improve the assembly, and why or why not?

Task 1.5: PacBio only assembly using canu

- perform the assembly as for the course, using the `fastq` file with subreads
- call this `canu_pacbio`

Questions you need to answer

- how does the canu assembly compare to SPADES?

Part 2: reference free assembly evaluation

- for this part, use the following assemblies:
  - `velvet_pe+mp`, the paired end and mate pair velvet assembly
  - `spades_pe+mp`, the Illumina-only SPADES assembly

Task 2.1: Mapping

- map the paired end reads and mate pair reads to the scaffold files with BWA
- **PhD students**: for the velvet_pe+mp assembly, plot the insert size distribution using the IPython notebook, show the figures for both libraries and explain what it means.

Task 2.2: REAPR

- perform REAPR analysis for the assemblies, as per the course instructions

Task 2.3: Genome browser

- add the following to IGV genome browser for both assemblies:
  - assembled scaffolds
  - gap track (use `scaffoldgap2bed.py` first, NOTE 1 bp minimum gap size)
  - mapped reads
  - Reapr's `03.score.errors_nospaces.gff` file (after converting spaces to underscores as described in the tutorial)

Questions you need to answer

- for each assembly, how many places did REAPR decide to make a break?
- for the `velvet_pe+mp` assembly, show one or two screenshots of alignments at positions where REAPR decided to split a scaffold/contig. Explain why you agree/disagree with REAPR

Part 3: reference-based assembly evaluation

- for this part, use the following assemblies:
  - `velvet_pe+mp`, the paired end and mate pair velvet assembly
  - `spades_pe+mp`, the Illumina-only SPADES assembly
  - **PhD students only** `spades_hybrid`, the hybrid SPADES assembly
  - `canu_pacbio`, the PacBio canu assembly

Task 3.1: Reference comparison

- compare the assemblies to the reference sequence using `quast`
- the reference fasta file is
  `/data/exam/assembly/reference/CP005385.1_Meiothermus_ruber_DSM_1279.fna`
- the file with positions of genes in the reference is
  `/data/exam/assembly/reference/CP005385.1_Meiothermus_ruber_DSM_1279.gff`
- show the `report.html` file during the exam

Questions you need to answer

- was one assembly clearly better than the others when compared to the reference, and why or why not?
- given all of the above, which assembly is 'better'?

Good luck!

PS

IMPORTANT use a minimum gap length of 1 bp for all analysis (not the default from each tool).