BME1063 Homework1 report

潘修齐 2018522077

BME1063 Homework1 report
- Data Preparation
  - Data Overview
  - data regeneration
  - detailed parameters
  - DataQC
- Results
  - Time summary
  - mapping quality summary
  - Cumulative precision and recall
- Discussion
  - Cause of time differences
  - Mapping quality Score
  - Possible reasons for mistakes
  - Recommendations

Data Preparation

Data Overview

The fastq data is generated by the software dwgsim written by `Nils Homer`. Most of the parameters are set as default except a few. In order to simulate data similar to real Next Generation Sequencing (NGS) data, the sequencing error rate is set to 0.0007 to produce reads with acceptable quality scores. The random seed involved in the data generation is also specified to make the process reproduceable. In the end, 3 pair-end datasets have been generated based on human genome `hg38` with different read length, namely 50, 70 and 100 bp.

data regeneration

To regenerate the data, run pbs file on `PBS/HW1-1_DataSimulation/HW1-1_DataSimulation.pbs`

```
qsub PBS/HW1-1_DataSimulation/HW1-1_DataSimulation.pbs

# for personal computer, run as shell script:
# bash PBS/HW1-1_DataSimulation/HW1-1_DataSimulation.pbs
```

detailed parameters

For running `dwgsim` to generate the data, the parameters are set as follows:

parameter	value
error rate	0.0007 (quality score ~31.5)
number of reads	100000
random seed	1063 (course number)
outer distance between two ends for pairs	mean 500, stddev 50
mutation rate	0.1%
indel fraction	0.1
random DNA read fraction	5%
number of Ns in a read	maximum 0
stddev for base quality score	50

DataQC

In order to ensure that the quality of the data simulated are within acceptable range, fastqc was performed upon the 3 datasets generated. To regenerate QC process, run pbs script `PBS/HW1-2_FASTQC/HW1-2_FASTQC.pbs`.

```
qsub PBS/HW1-2_FASTQC/HW1-2_FASTQC.pbs

# for personal computers, run as shell script:
# bash PBS/HW1-2_FASTQC/HW1-2_FASTQC.pbs
```

The QC result can be found on `FinalReport/FASTQC`. According to the result, all the simulated datasets performs as good as a real-life high-quality sequencing data, laying the foundation for further analysis.

Results

2 different software, namely `BWA` and `Bowtie2`, is used to align the simulated data to reference genome `hg38`. For optimal running efficiency, both software uses a pre-index strategy to speed up the alignment process. For `BWA`, the index is generated with resources up to 10-core, 32 Gb ram on HPC. For `bowtie2`, the pre-made index file for the same reference genome was downloaded from the official website.

Time summary

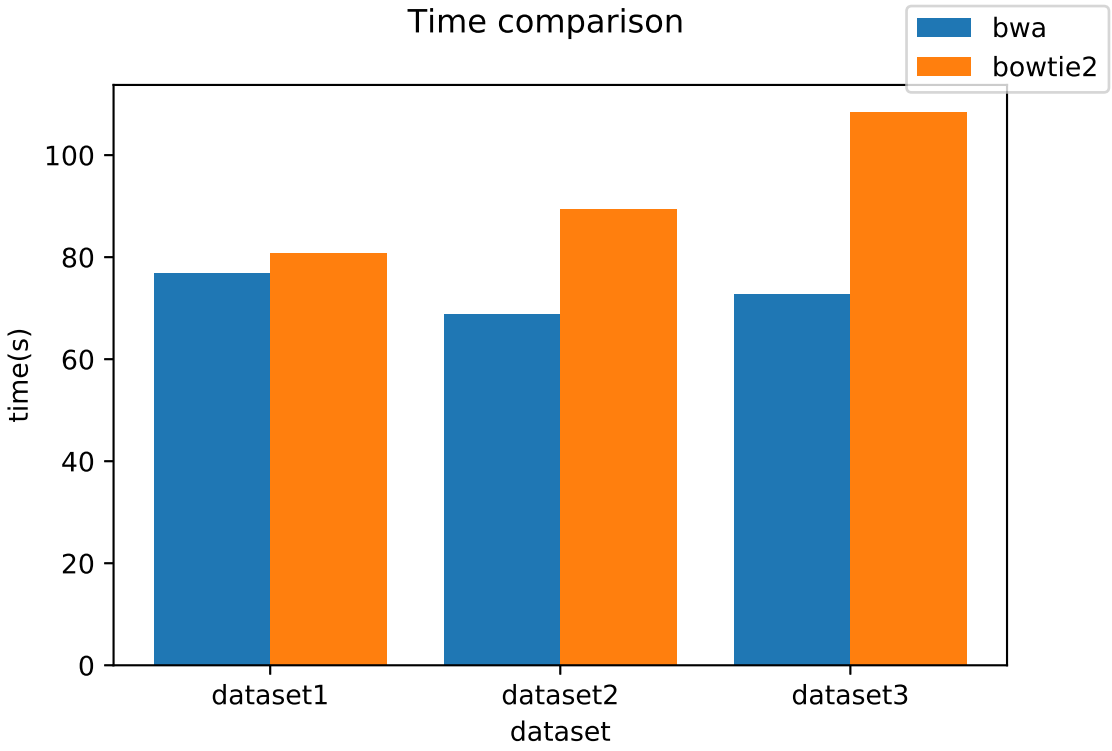
Running times for the two software are measured using Linux `time` function and are extracted by script `script/1-time_summary.sh`

```
bash script/1-time_summary.sh
```

The running time summary can be found in `FinalReport/time_summary`. According to the summary, the running times (user time) of two software are as follows:

dataset	BWA time	Bowtie2 time
dataset1	1m17s	1m21s
dataset2	1m9s	1m29s
dataset3	1m13s	1m48s

A visualization can be generated as follows using code from `script/2-vidualization_analyzation.ipynb`.



As is shown above, BWA and bowtie2 have a similar running time for dataset1. However, as the read length getting longer, BWA running time stays constant while bowtie2 require more time to perform alignment.

mapping quality summary

The mapping quality information is summarized by calling `dwgsim_eval`, which is a component of `dwgsim`. Tab-splitted summary files will be generated on `FinalReport/mapping_summary` by running `script/3-map_quality_summary.sh`. The further analyzation and visualization is done by jupyter notebook `script/2-vidualization_analyzation.ipynb`. To assess the data, precision, recall and F-score is defined as follows:

$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{TP + FN}$$
$$Fscore = \frac{2 * Precision * Recall}{Precision + Recall}$$

The result is summarized in the following table:

software	dataset	precision	recall	F-score
bwa	dataset0	0.930225	1.0	0.963852
bowtie2	dataset1	0.909904	0.999397	0.952553
bwa	dataset1	0.915926	1.0	0.956119
bowtie2	dataset2	0.883572	0.998894	0.937701
bwa	dataset2	0.923573	1.0	0.960268
bowtie2	dataset3	0.900112	0.999278	0.9471060

In terms of performance, BWA tends to do better than Bowtie2 in general.

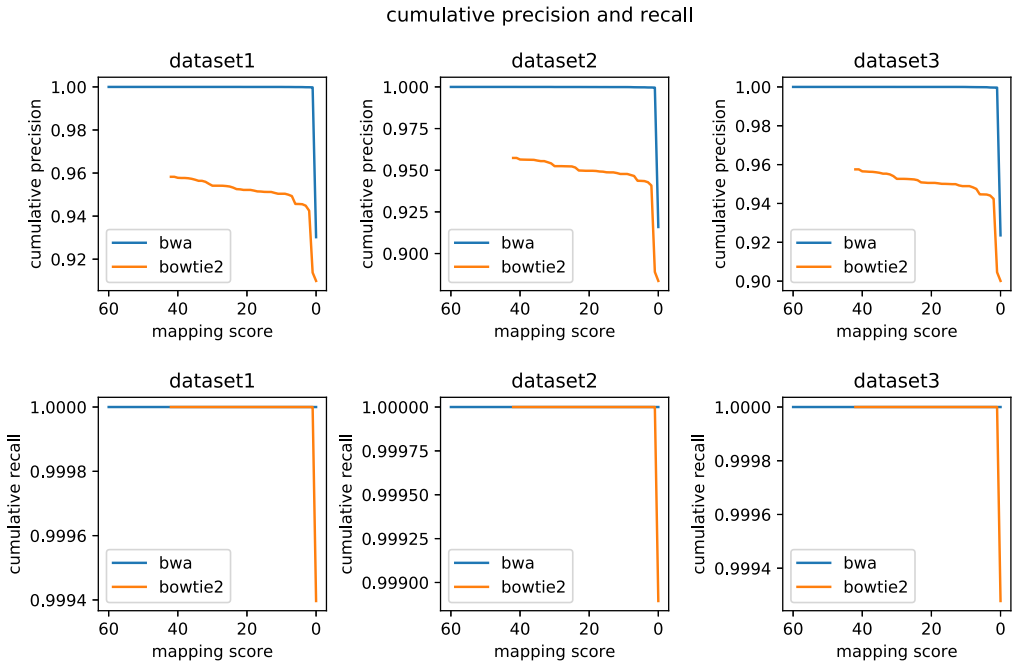
Cumulative precision and recall

Since both software provide with mapping a mapping quality score, it is informative to see the mapping quality with mapping quality score above some threshold. To achieve that goal, a cumulative precision and recall was defined as follows:

$$\text{Cumulative precision} = \frac{\text{TP considering cases with score higher than threshold}}{\text{TP + FP considering cases with score higher than threshold}}$$

$$\text{Cumulative recall} = \frac{\text{TP considering cases with score higher than threshold}}{\text{TP + FN considering cases with score higher than threshold}}$$

The following figure visualizes the cumulative precision and recall of the result generated by BWA and Bowtie. Note that different software has different quality scoring strategy, and thus quality scores produced by different software cannot be compared with each other.



Discussion

Cause of time differences

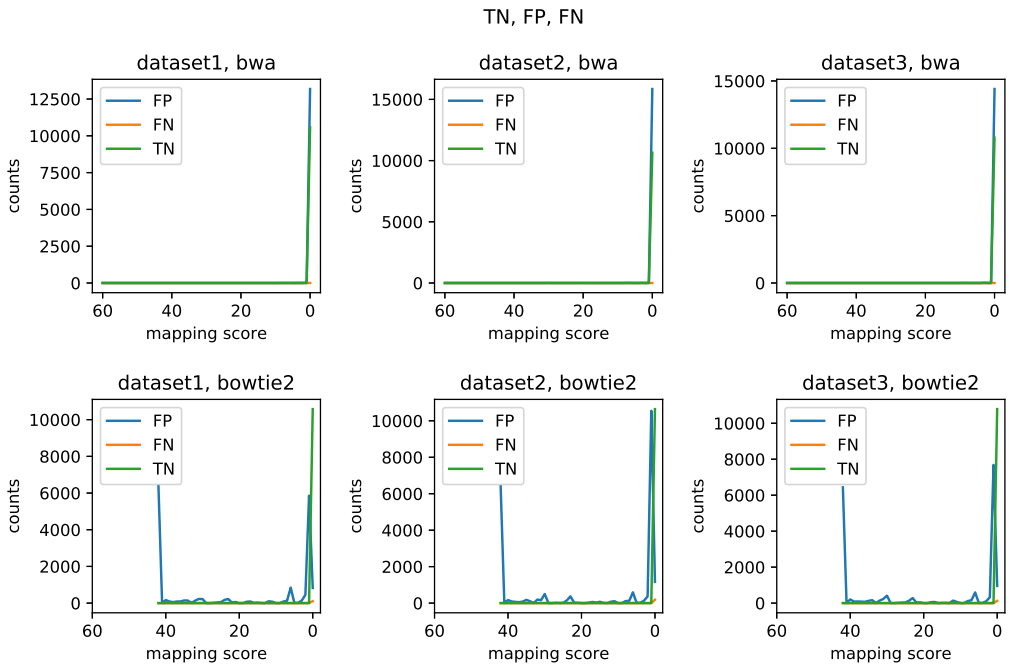
As we can see, BWA runs with constant time with 3 different datasets while bowtie2's running time varies between different datasets. Since the only difference between the datasets is the sequencing read length, we can predict that Bowtie2 requires longer running time for longer sequencing reads while BWA do not. This phenomenon is caused by the algorithms applied by the two software.

For bowtie2, a 4-steps alignment process is implemented. The first step for Bowtie2 to align a single read is generating "seed" substrings for every 10 bp of the query string, and thus the number of seeds is proportional to the length of query read. After that, Burrows Wheeler alignment was performed to align the seeds to the reference genome, followed by extension using dynamic programming algorithm. Because of the increasing number of seeds with query length and quadratic complexity of dynamic programming algorithm, the time spent will get longer for mapping longer reads, being $O(mnN)$. (m being the read length, n being number of reads and N being the reference size)

For BWA, the seeding process will create similar number of seed regardless of the query length and alignment is based on Burrows-Wheeler Transform. Starting from the whole reference genome, the search space for BWA get smaller by each round. In average, for every round, 3 fourths of the reference genome is eliminated, leading to a time complexity of $O(nlog_4(N))$ for each reads, N being the size of the Genome and n being the number of reads. Since $4^50 \approx 10^{30}$ is a large number, well exceeding the size of human reference genome, the alignment will terminate before running out of read length. As a result, time consumption for BWA alignment is determined by the reference genome size and is unrelated with read length.

Mapping quality Score

As we can observe by `cumulative precision and recall`, bad mappings are clustered at around 0 score, especially for BWA. In fact, If we can plot FP, TN, FN against the mapping quality score, we will find that those events all happen with relatively low quality score, proving the scoring system reasonable.



Possible reasons for mistakes

The most common mistake is False Positive, and as is recorded in `script/2-vidualization_analyzation.ipynb`, there is no random sequences mapped, thus the most common mistake being mis-mapping. It is found that mistakes can happen when the read comes from repetative regions.

we take one mis-mapped read as an example:

`chr1_122619406_122619831_0_1_0_0_0:0:0_0:0:0_4b9`

This read is from dataset3, originating in chromosome 1, position 122619406. It is mapped to position 25024848 at chromosome 19 by bwa and position 123137299 at chromosome 1 by bowtie2.

The raw sequence is:
`TATTCACCTCACCGATTTGAACGATCCTTTACACAGAGCAGACTTGAAACACTCTTTTTGTGGAATTTGCAAGTGGAGATTTCAGCCGCTTTGAGGTCAA`

By searching the sequence using Human BLAT Search provided by UCSC, the result is as follows:

ACTIONS	QUERY	SCORE	START	END	QSIZE	IDENTITY	CHROM	STRAND	START	END	SPAN
browser details	YourSeq	100	1	100	100	100.0%	chr5	+	47425343	47425442	100
browser details	YourSeq	100	1	100	100	100.0%	chr19	+	25024848	25024947	100
browser details	YourSeq	100	1	100	100	100.0%	chr1	+	122619406	122619505	100
browser details	YourSeq	96	1	100	100	98.0%	chr5	+	47611980	47612079	100
browser details	YourSeq	96	1	100	100	98.0%	chr19	+	25211485	25211584	100
browser details	YourSeq	96	1	100	100	98.0%	chr1	+	124323375	124323474	100
browser details	YourSeq	96	1	100	100	98.0%	chr1	+	122991673	122991772	100
browser details	YourSeq	96	1	100	100	98.0%	chr1	+	122508552	122508651	100
browser details	YourSeq	94	1	100	100	97.0%	chr1	+	123523209	123523308	100
browser details	YourSeq	94	1	100	100	97.0%	chr1	+	122947148	122947247	100
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	122651030	122651113	84
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	122633010	122633093	84
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	122629270	122629353	84
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	122628590	122628673	84
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	122583035	122583118	84
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	122572495	122572578	84
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	122504485	122504568	84
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	121967850	121967933	84
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	121765855	121765938	84
browser details	YourSeq	84	17	100	100	100.0%	chr1	+	121764840	121764923	84

The sequence actually comes from a satellite DNA region, ALR/Alpha. It is a highly repetitive region in the genome and makes it difficult for aligners to map correctly. For such cases, one can adjust parameters to force the aligner to output all the possible maps. For BWA, use `BWA aln -N` and for Bowtie2, use `bowtie2 -a`.

Recommendations

In terms of time consumption, `BWA` have a better performance for reads longer than 50bp. In terms of mapping quality, `BWA` also more accurate than `Bowtie2`. As a result, `BWA` is more recommended in terms of short reads alignment.