Genome Assembly Lab Part1 – Data Preprocessing

BCB 5250 Introduction to Bioinformatics II

Spring 2020

Tae-Hyuk (Ted) Ahn
Department of Computer Science
Program of Bioinformatics and Computational Biology
Saint Louis University

SAINT LOUIS
UNIVERSITY
EST. 1818

Lab: Genome Assembly Lab

- Quality control with FastQC
- Data trimming and error correction (optional)
- Genome assembly with Velvet
- Genome assembly with Spades (Homework)
- Validate the assembly with Quast

Genome assembly with Velvet

- For this tutorial, we have a set of reads from an imaginary *Staphylococcus aureus* bacterium with a miniature genome (197,394 bp).
- Our mutant strain read set was sequenced with the whole genome shotgun method, using an Illumina DNA sequencing instrument.
- From these reads, we would like to rebuild our imaginary *Staphylococcus aureus* bacterium via a *de novo* assembly of a short read set using the Velvet assembler.

Data

- Connect to hopper.slu.edu
- Make your directory for this lab as like: (what is –p option?)

```
$ mkdir -p Course/bcb5250/labs/genome_assembly_lab
$ cd Course/bcb5250/labs/genome_assembly_lab
```

- Download the data

```
$ mkdir data
$ cd data
$ wget https://zenodo.org/record/582600/files/mutant_R1.fastq
$ wget https://zenodo.org/record/582600/files/mutant_R2.fastq
```

PS. If the network is slow, copy the files from hopper.slu.edu:/ /public/ahnt/courses/bcb5250/genome_assembly_lab/data/ to your local directory.

Inspect the content of a file.

- What are four key features of a FASTQ file?
- What is the main difference between a FASTQ and a FASTA file?
- How long are the sequences?
- What is the average coverage of the genome, given our imaginary *Staphylococcus aureus* bacterium has a genome of 197,394 bp?

Evaluate the input reads

Before doing any assembly, the first questions you should ask about your input reads include:

- What is the coverage of my genome?
- How good is my read set?
- Do I need to ask for a new sequencing run?
- Is it suitable for the analysis I need to do?

Quality Control of the Data

- FastQC: Quality control check

  https://www.bioinformatics.babraham.ac.uk/projects/fastqc/

  - FastQC aims to provide a simple way to do some quality control checks on raw sequence data coming from high throughput sequencing pipelines. It provides a modular set of analyses which you can use to give a quick impression of whether your data has any problems of which you should be aware before doing any further analysis.

FastQC

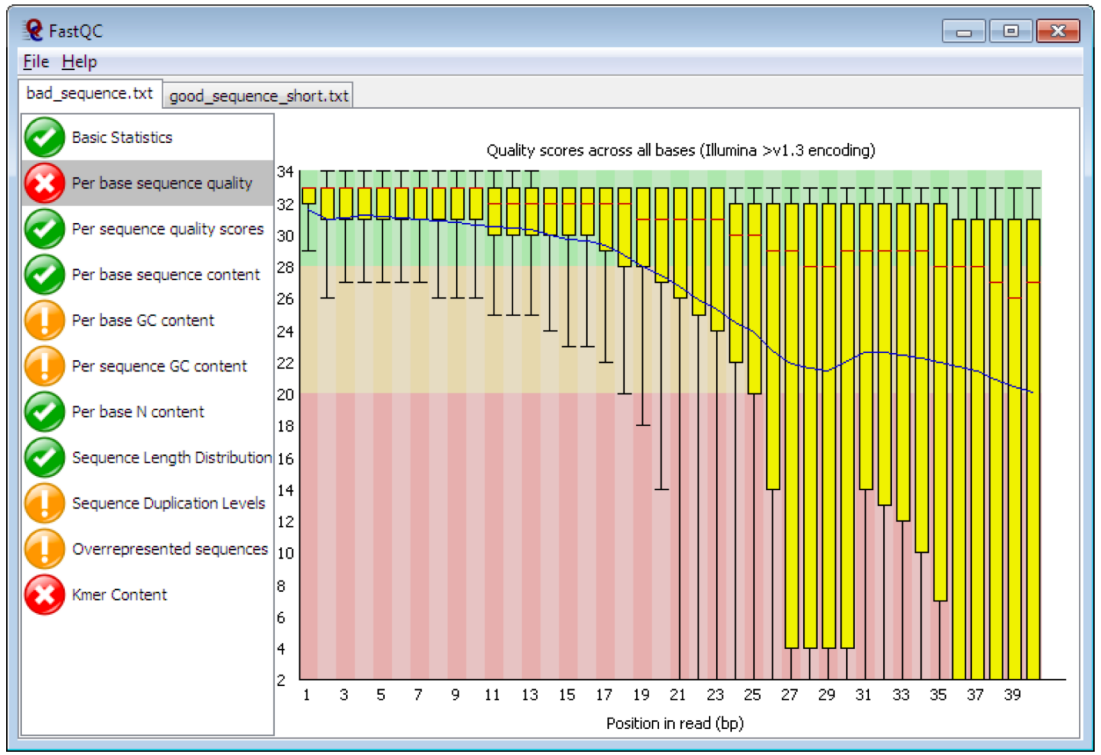
Function	A quality control tool for high throughput sequence data.
Language	Java
Requirements	A suitable Java Runtime Environment The Picard BAM/SAM Libraries (included in download)
Code Maturity	Stable. Mature code, but feedback is appreciated.
Code Released	Yes, under GPL v3 or later.
Initial Contact	Simon Andrews

Download Now

Quality Control of the Data

The main functions of FastQC are

- Import of data from BAM, SAM or FastQ files (any variant)
- Providing a quick overview to tell you in which areas there may be problems
- Summary graphs and tables to quickly assess your data
- Export of results to an HTML based permanent report
- Offline operation to allow automated generation of reports without running the interactive application



Quality Control of the Data

- Run FastQC

```
$ fastqc -h
$ fastqc mutant_R1.fastq mutant_R2.fastq
```

- Check the html results
  - Copy the html files to you laptop and open it

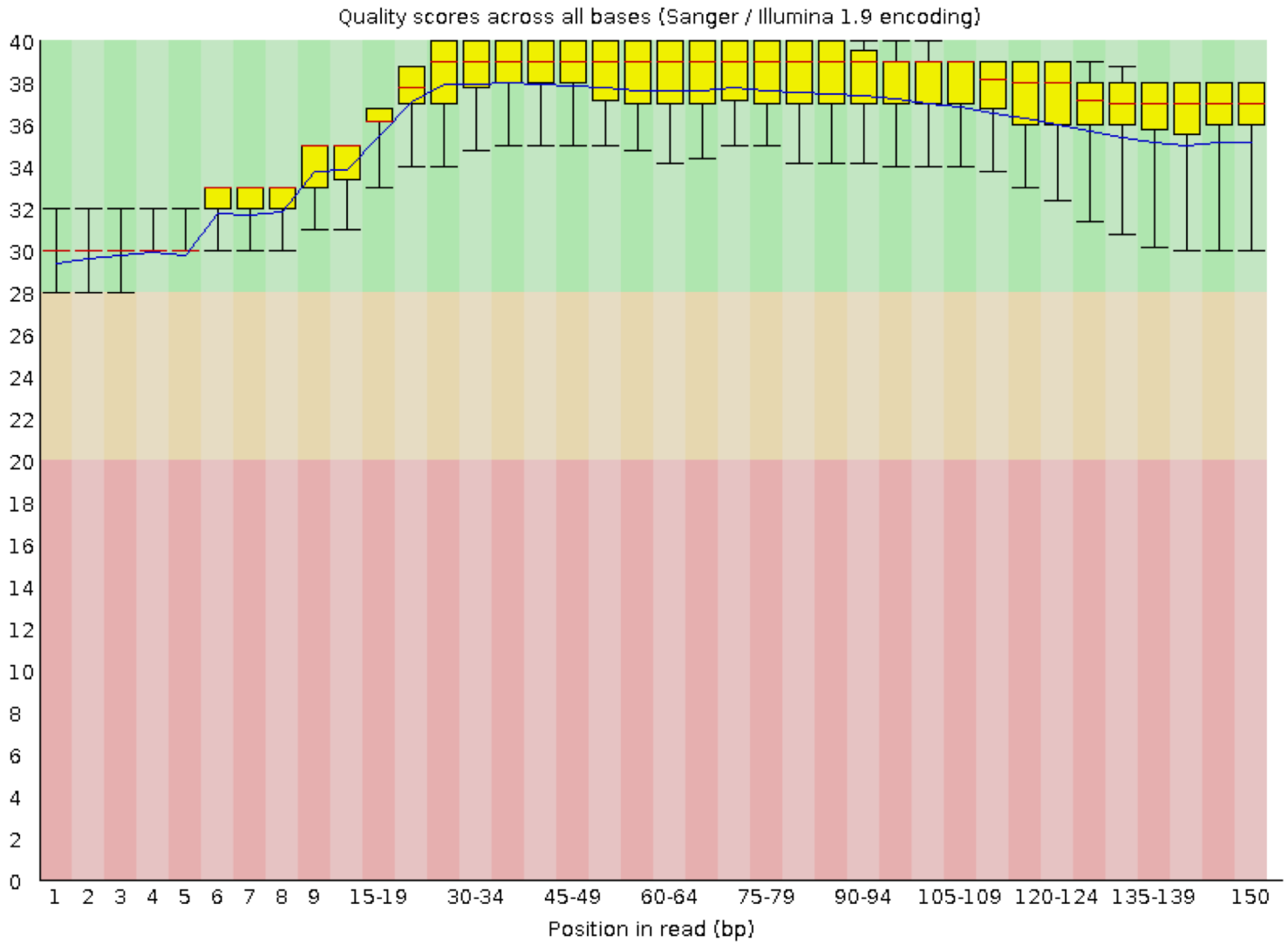
FASTQC Report

Basic Statistics

Measure	Value
Filename	mutant_R1.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	12480
Sequences flagged as poor quality	0
Sequence length	150
%GC	33

FASTQC Report

Per base sequence quality

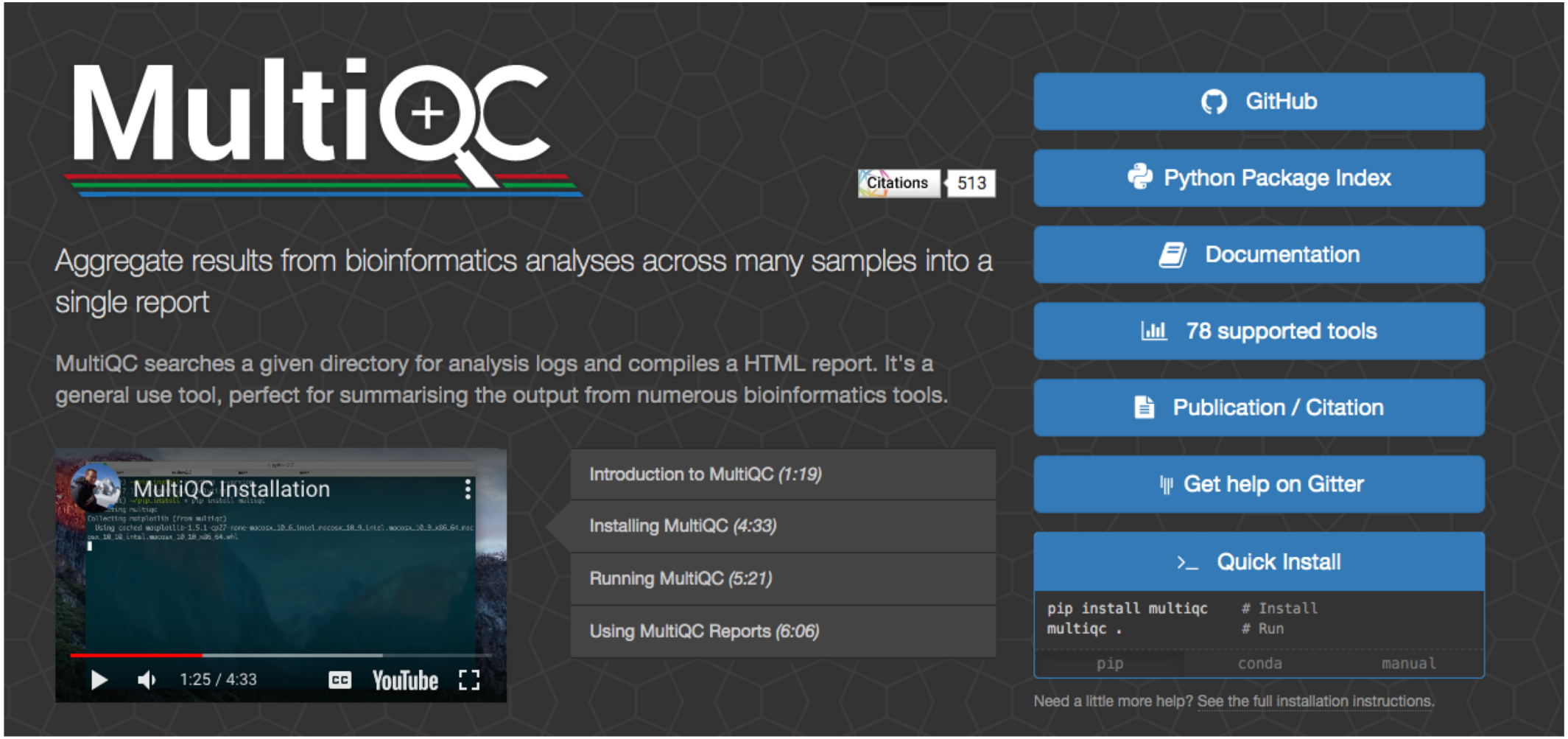


Merge FASTQC Reports

- Let us use it to evaluate the quality of our FASTQ files and combine the results with MultiQC.

MultiQC

- https://multiqc.info/



How to install software by yourself?

- Some software tools are not pre-installed and you need to install them by yourself.
- Let us install MultiQC and run it.

$ multiqc .

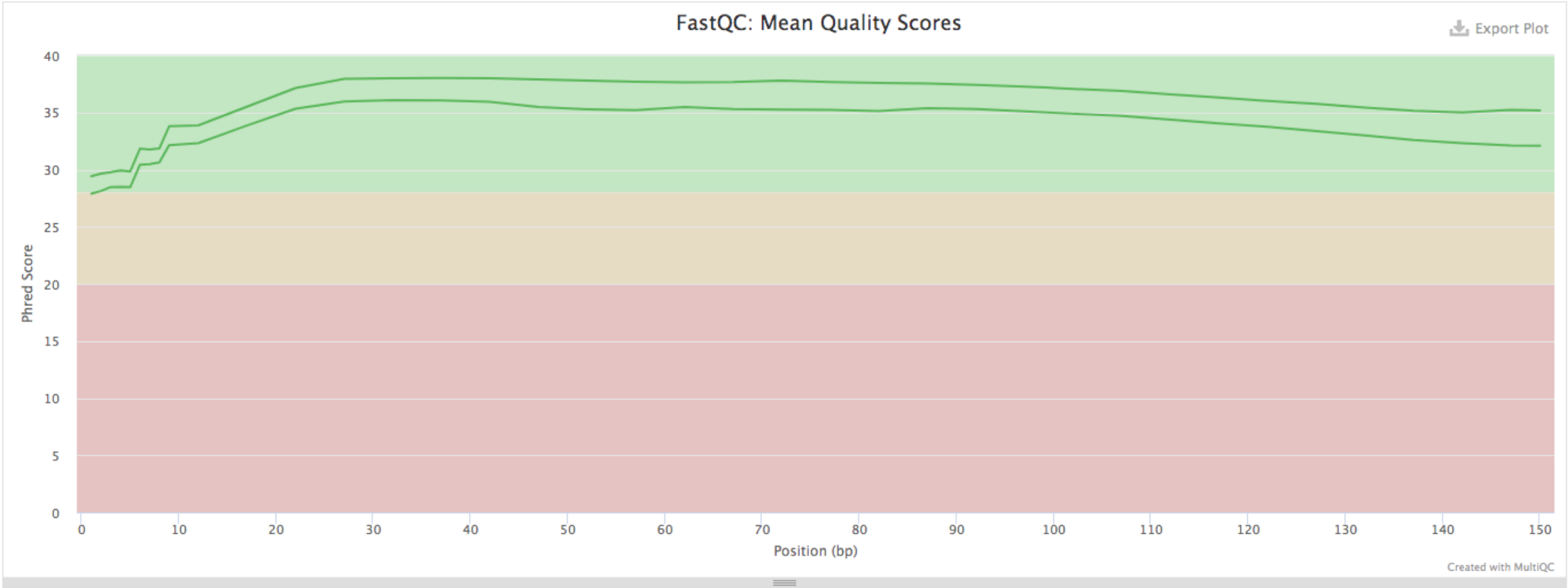
MultiQC Report

Sequence Quality Histograms 2

Help

The mean quality value across each base position in the read.

Y-Limits: on



Questions?

- What does the y-axis represent?
- Why is the quality score decreasing across the length of the reads?

- Study Phred quality score by yourself!
  - Simple tutorial: https://en.wikipedia.org/wiki/Phred_quality_score
  - It will be a Quiz question!

Trim, Filter, and Error Collection

Trim and filtering:

- FASTX Tool Kit: http://hannonlab.cshl.edu/fastx_toolkit/
- Trimmomatic: http://www.usadellab.org/cms/?page=trimmomatic
- Tim Galore: http://www.bioinformatics.babraham.ac.uk/projects/trim_galore/
- fastp: https://github.com/OpenGene/fastp
- BBTools – BBDuk: https://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/bbduk-guide/

Error correction:

- BBTools – Tadpole: https://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/tadpole-guide/
- https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-017-1784-8
- For long sequencing, check the recently published papers and archive such as http://www.pacb.com/asset_tags/error-correction/

Error correction can lead assembly problem!

Conclusions

The performance of different EC tools was compared using two approaches: the ability of EC tools to correct sequencing errors in Illumina data, and the effects of those corrections on the resulting *de novo* genome assembly quality. We found that EC tools correct a significant fraction of sequencing errors. However, state-of-the-art Illumina assemblers do not always appear to benefit from this. The assembly results for eight different datasets with SPAdes and DISCOVAR show that the prior application of EC tools often does not lead to a significant increase in NGA50, and in fact may result in a lower NGA50. Many erroneous corrections occur in regions that have low read coverage and in the vicinity of highly frequent repeats. Due to the low coverage, error correction tools incorrectly assume the presence of sequencing errors. The repeated elements on the other hand cause erroneous substitutions to be applied. A too aggressive and/or inconsistent transformation of such reads in such region may lead to loss of information from which no recovery is possible during the assembly process. This inevitably leads to an increased assembly fragmentation. Additionally, the prior use of EC tools does not lead to a major decrease in overall runtime and/or memory requirements compared with the assembly from uncorrected data.

From a methodological point of view, multiple sequence alignment (MSA) based methods might have an advantage over methods that operate on isolated *k*-mers. MSA-based methods take multiple reads into account when applying substitutions and hence appear to make more consistent corrections across overlapping reads.

Extra: How to open report html file without remote copy?

Linux or Mac:

Instead of

$ ssh YourAccount@hopper.slu.edu

Using –X option (

$ ssh –X YourAccount@hopper.slu.edu

Windows:

Search “X Server for Windows”

Eg: Cygwin, Xming, …