Introduction to NGS data analysis

Richard Smith-Unna

1. Rigour and reproducibility

2. Getting your read data

3. What your data looks like: the FASTQ format

4. Organising the raw data

5. Read data quality checking

6. Quality improvement

7. Getting hold of a reference

8. Aligning reads to the reference

9. Application-specific workflows

1. Rigour and reproducibility

- Bioinformatics is *science*

- Keep a (digital) lab book

- Diligently record protocols:

  - software versions

  - commands

  - settings

  - file manipulations

  - numerical conversions

- Publish your code

- Data driven discovery is valid science, but requires self control

2. Getting your read data

- FTP or HTTP transfer (wget or curl)

- Might require a special program or plugin (e.g. Aspera)

- Files are compressed FASTQ, separate files for left and right pairs

- Verify file integrity after download (md5sum)

- Pay attention to download expiry dates

- Backup

Subject: Re: Re: F14FTSEUHT01 ▇▇▇ data release
Date: 2014-07-24 11:01
From: shirley xiaoxi guo <shirley.guo@bgitechsolutions.com>
To: ▇▇▇▇▇▇▇@cam.ac.uk>

Dear ▇▇▇,

Please download data as below:

Link: http://cdts.genomics.hk/customerSupport/login.xhtml [1]
Account: 20140723F14FTSEUHT01▇▇▇
Password: CHLhtbT201▇▇▇

3. What your data looks like: the FASTQ format

```
@HWUSI-EAS100R:6:73:941:1973#0/1
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

sequence identifier

| HWUSI-EAS100R	the unique instrument name
6	flowcell lane
73	tile number within the flowcell lane
941	'x'-coordinate of the cluster within the tile
1973	'y'-coordinate of the cluster within the tile
#0	index number for a multiplexed sample (0 for no indexing)
/1	the member of a pair, /1 or /2 *(paired-end or mate-pair reads only)*

base quality

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
```

4. Organising the raw data

- if samples are multiplexed, map barcodes to samples

- may require splitting, e.g. with **fastx-toolkit barcode splitter**

- organise according to condition and sample

- give files machine-parseable names

  - rice_wet_rep1_left.fq.gz
  - rice_wet_rep1_right.fq.gz
  - rice_wet_rep2_left.fq.gz
  - rice_wet_rep2_right.fq.gz
  - rice_dry_rep1_left.fq.gz
  - rice_dry_rep1_right.fq.gz
  - rice_dry_rep2_left.fq.gz
  - rice_dry_rep2_rightt.fq.gz

5. Read data quality checking

http://www.bioinformatics.babraham.ac.uk/projects/fastqc
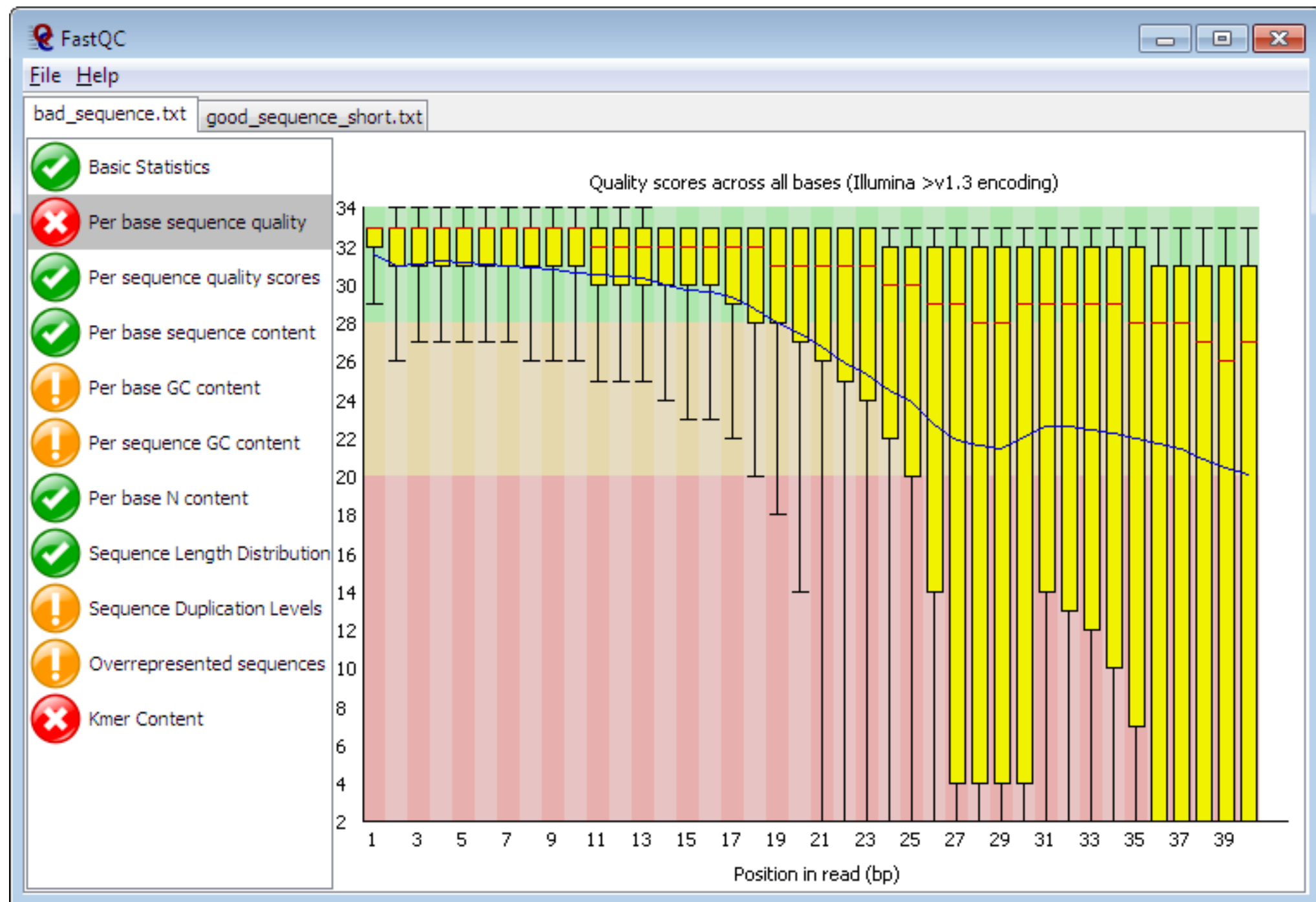
5. Read data quality checking
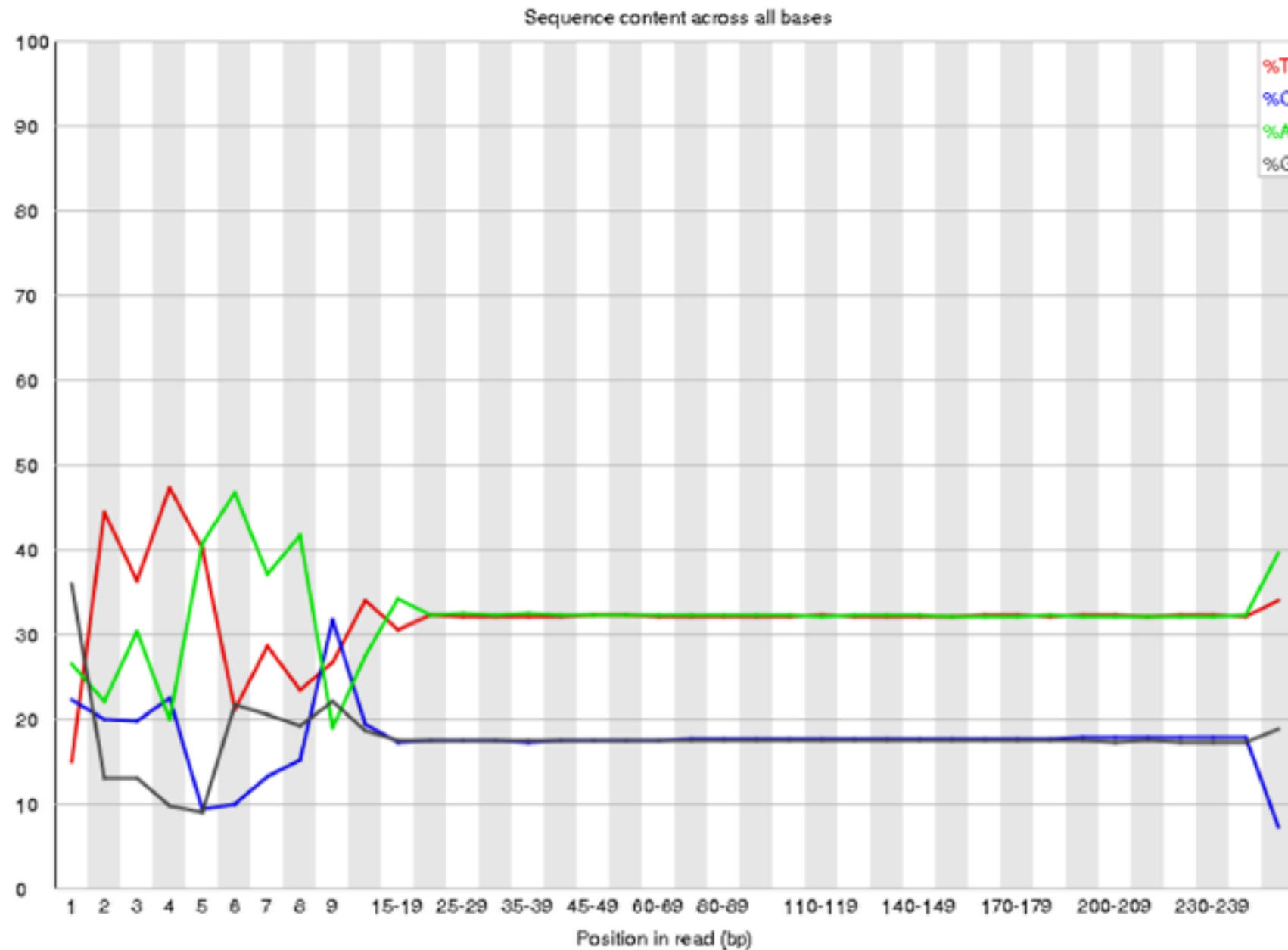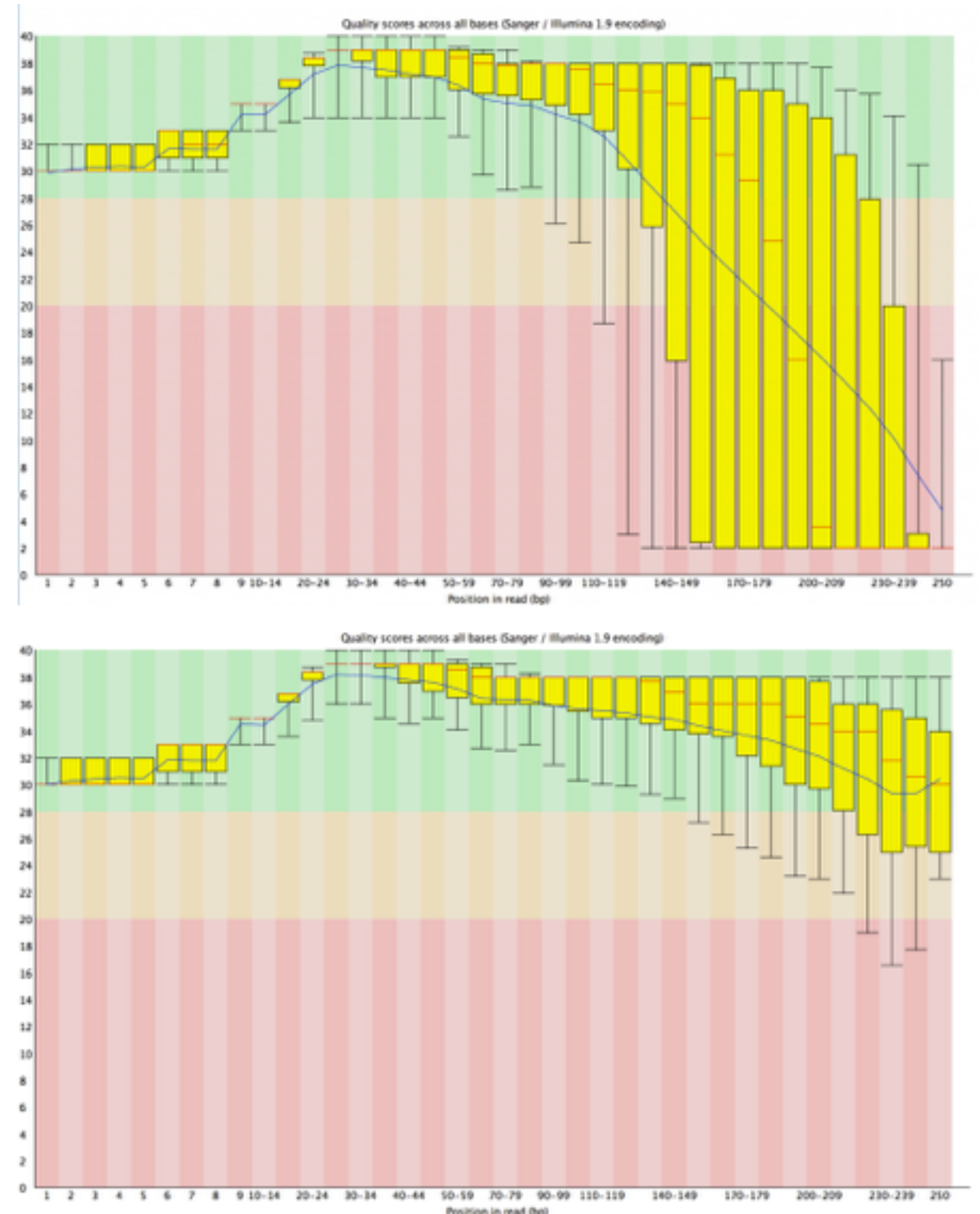

Sequence content (%)

6. Quality improvement

common approaches:

- trimming adapters

- trimming low-quality bases

- trimming to a fixed length

- discarding low-quality reads

tools:

- trimmomatic

- fastx-toolkit

7. Getting hold of a reference

genome or transcriptome?

- genome for:

  - variant calling

  - DNase-seq

  - CHiP-seq

- transcriptome for:

  - RNAseq

- FASTA format

>Seq1 [organism=Carpodacus mexicanus] actin (act) mRNA, partial cds
CCTTTATCTAATCTTTGGAGCATGAGCTGGCATAGTTGGAACCGCCCTCAGCCT
CCTCATCCGTGCAGAACTTGGACAACCTGGAACTCTTCTAGGAGACGACCAAAT
TTACAATGTAATCGTCACTGCCCACGCCTTCGTAATAATTTTCTTTATAGTAATAC
CAATCATGATCGGTGGTTTCGGAAACTGACTAGTCCCACTCATAATCGGCGCCC
CCGACATAGCATTCCCCCGTATAAACAACATAAGCTTCTGACTACTTCCCCCATC
ATTTCTTTTACTTCTAGCATCCTCCACAGTAGAAGCTGGAGCAGGAACAGGGTG
AACAGTATATCCCCCTCTCGCTGGTAACCTAGCCCATGCCGGTGCTTCAGTAGA
CCTAGCCATCTTCTCCCTCCACTTAGCAGGTGTTTCCTCTATCCTAGGTGCTATT
AACTTTATTACAACCGCCATCAACATAAAACCCCCAACCCTCTCCCAATACCAAA
CCCCCCTATTCGTATGATCAGTCCTTATTACCGCCGTCCTTCTCCTACTCTCTCTC
CCAGTCCTCGCTGCTGGCATTACTATACTACTAACAGACCGAAACCTAAACACTA
CGTTCTTTGACCCAGCTGGAGGA

7. Getting hold of a reference

- sources of existing references:

  - ENSEMBL
    http://ensembl.org

  - JGI
    http://genome.jgi.doe.gov

  - EBI genomes
    http://www.ebi.ac.uk/
    genomes/bacteria.html

- assembling your own:

  - covered tomorrow

8. Aligning reads to the reference

- Estimate the original source location of each read

- Simon will cover this in the next lecture

9. Application-specific workflows

- Usually starts by converting read alignments into some numerical summary data (e.g. expression counts, peak calls)

- Running programs from the command-line

- Using specialist packages in a programming language like R (http://bioconductor.org), Python or Ruby (http://biogems.info)

- Writing your own analysis code

- Performing statistical analysis on alignment summary data

- Normalisation: accounting for different sample sizes

- Storytelling and plotting