• Support service for bioinformatics
  – Academic – Babraham Institute
  – Commercial – Consultancy

• Support BI Sequencing Facility
  – MiSeq/ HiSeq/ NextSeq-based sequencing service
  – Data Management / Processing / Analysis
Interests in QC

- QC packages
- Application specific QC
- Data visualisation QC

[Images of QC tools and software]
An example of why QC is important...

- Single Cell RNA-Seq
  - Each dot is a cell
  - An outgroup is clearly visible
  - What is it?
## Genes for PC2 (85 total)

<table>
<thead>
<tr>
<th>Gene</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arhgef4</td>
<td>Rho guanine nucleotide exchange factor (GEF) 4</td>
</tr>
<tr>
<td>Cflar</td>
<td>CASP8 and FADD-like apoptosis regulator</td>
</tr>
<tr>
<td>Als2</td>
<td>amyotrophic lateral sclerosis 2 (juvenile) homolog (human)</td>
</tr>
<tr>
<td>Cxcr2</td>
<td>chemokine (C-X-C motif) receptor 2</td>
</tr>
<tr>
<td>Col4a3</td>
<td>collagen, type IV, alpha 3</td>
</tr>
<tr>
<td>Sag</td>
<td>retinal S-antigen</td>
</tr>
<tr>
<td>Gpr35</td>
<td>G protein-coupled receptor 35</td>
</tr>
<tr>
<td>Acmsd</td>
<td>amino carboxymuconate semialdehyde decarboxylase</td>
</tr>
<tr>
<td>Qsox1</td>
<td>quiescin Q6 sulfhydryl oxidase 1</td>
</tr>
<tr>
<td>9430070013Rik</td>
<td>RIKEN cDNA 9430070013 gene</td>
</tr>
<tr>
<td>Mrps14</td>
<td>mitochondrial ribosomal protein S14</td>
</tr>
<tr>
<td>Scyl3</td>
<td>SCY1-like 3 (S. cerevisiae)</td>
</tr>
<tr>
<td>Ildr2</td>
<td>immunoglobulin-like domain containing receptor 2</td>
</tr>
<tr>
<td>Atp1a2</td>
<td>ATPase, Na+/K+ transporting, alpha 2 polypeptide</td>
</tr>
<tr>
<td>Slamf8</td>
<td>SLAM family member 8</td>
</tr>
<tr>
<td>Wdr38</td>
<td>WD repeat domain 38</td>
</tr>
<tr>
<td>Exd1</td>
<td>exonuclease 3'-5' domain containing 1</td>
</tr>
<tr>
<td>Serf2</td>
<td>small EDRK-rich factor 2</td>
</tr>
</tbody>
</table>
Coverage of Raw Data

Normal Gene

PCA Gene
Conclusion: The separation in the original graph was a technical artefact of no biological interest. If we'd published this it would have mislead others. Even if we find it we've wasted some time and effort.
What is the point of QC?

• Technical problems don't cause pipelines to fail
• Technical problems don't prevent hits being generated
• Technical hits often look biologically real
• Unexpected, interesting effects can easily be missed
• Finding problems through follow-on work is slow and expensive!

• QC saves you time and effort! (and money)
Course Structure

• How does Illumina Sequencing work?

• What can QC tell us?
  – QC Software
  – Universal metrics
  – Library Dependent metrics
  – Consistency

• Putting QC into Practice
How Does Illumina Sequencing Work?
Illumina Sequencing: An Overview

- **Library Generation**
  - Adapter
  - Insert
  - Adapter

- **Attach to Flow Cell**

- **Sequence by Synthesis (SBS)**
  - DNAP

- **Sequencing Reads**
Microscope

Fridge

Compartment

Pumps
Flow Cell
Illumina Sequencing: An Overview

Library Generation

Attach to Flow Cell

Sequence by Synthesis (SBS)

Sequencing Reads

Adapter Insert Adapter
SBS For a Single Molecule

G A T T T C A G
SBS For a Single Molecule

C

G

A

T

T

C

A

G
SBS For a Single Molecule

1 Cycle
SBS For a Single Molecule

CTA\textbullet\textbullet\textbullet\textbullet\textbullet\textbullet\textbullet\textbullet\textbullet AGCTC\textbullet\textbullet\textbullet\textbullet\textbullet\textbullet\textbullet\textbullet\textbullet

GATTTCACAG

7 Cycles \hspace{1cm} \ldots \text{Read length}
Comparing Chemistry

Signal → Image → Base Call

4 Channel Chemistry

2 Channel Chemistry

1 Channel Chemistry
Comparing Chemistry

No Signal is interpreted as G
Detecting a Signal

- One Molecule isn’t Enough
- Amplify to generate Cluster
- Multiple Clusters on a Flow Cell
Flow Cell Imaging

Flow in

Image

Flow out

Image

Lanes

Swaths

Tiles
Real Illumina Sequence Data
Creating Clusters

Non Patterned Flow Cell

Patterned Flow Cell

Random Clusters

Pre-defined Clusters
Creating Clusters: Patterned Flow Cells

Single molecule attaches in a nano well → Recombinase Polymerase Bridge Amplification → Cluster of identical (ish) molecules created in a nanowell

Amplification is faster than seeding
Good and Bad things about clusters

Good
• Generates large signal
• Is robust to random mistakes
• Needs a small amount of starting material

Bad
• Bridging limits length
• Molecules in a cluster get out of sync
  – 2 bases added
  – No bases added
  – Reaction stalls
• Can get mixed signals if clusters overlap (non-patterned)
• Can get re-seeding (patterned)
• Can get index hopping (patterned)
Different sequencers, same chemistry

<table>
<thead>
<tr>
<th>Sequencer</th>
<th>Number of lanes</th>
<th>Reads per lane</th>
<th>Max read length</th>
<th>Dyes</th>
</tr>
</thead>
<tbody>
<tr>
<td>iSeq 100</td>
<td>1</td>
<td>~4 million</td>
<td>150bp</td>
<td>1</td>
</tr>
<tr>
<td>MiniSeq</td>
<td>1</td>
<td>~7 million</td>
<td>150bp</td>
<td>2</td>
</tr>
<tr>
<td>MiSeq</td>
<td>1</td>
<td>~20 million</td>
<td>300bp</td>
<td>4</td>
</tr>
<tr>
<td>NextSeq</td>
<td>1</td>
<td>~400 million</td>
<td>150bp</td>
<td>2</td>
</tr>
<tr>
<td>HiSeq 2xxx</td>
<td>16</td>
<td>~200 million</td>
<td>150bp</td>
<td>4</td>
</tr>
<tr>
<td>HiSeq 4xxx</td>
<td>16</td>
<td>~300 million</td>
<td>150bp</td>
<td>4</td>
</tr>
<tr>
<td>NovaSeq</td>
<td>8</td>
<td>~2.5 billion</td>
<td>150bp</td>
<td>2</td>
</tr>
</tbody>
</table>
Illumina Sequencing: An Overview

Library Generation

Attach to Flow Cell

Sequence by Synthesis (SBS)

Sequencing Reads
What Reads Do You Get

Single End Run (one fastq file)

Paired End Run (two fastq files)

Used to separate libraries – often don't see the fastq file

Barcode Read
FastQ Format Data

@HWUSI-EAS611:34:6669YAAXX:1:1:5069:1159 1:N:0:
TCGATAATACCGTTTTTTTCCGTTTGATGTTGATACCATT
+
IIHIIHIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII

@HWUSI-EAS611:34:6669YAAXX:1:1:5243:1158 1:N:0:
TATCTGTAGATTTCACAGACTCAAATGTAAATATGCAGAG
+
DF=DBD<BBFGGGGGGGGGBD@GGGD4@CA3CGG>DDD:D,B

@HWUSI-EAS611:34:6669YAAXX:1:1:5266:1162 1:N:0:
GGAGGAAGTATCACTTCCTTGCCTGCCTCCTGGGGCCT
+
:GBGGGGGGGGGGGGDGDGGDDEDDGGGGGGGDHDDHGHGHGBGG:GG
A single FastQ Entry

@HWUSI-EAS611:34:6669YAAXX:1:1:5266:1162 1:N:0:GGAGGAAGTATCACTTCCTGCTGCCTCCTCCTCTGGGGCCT
+:GBGGGGGGGGDGDEGDDGGGDDGGDEHHGHHGBGG:GG

1. Header - starts with @
2. Base calls (can include N or IUPAC codes)
3. Mid-line - starts with + usually empty
4. Quality scores (= Phred Scores)
Illumina Header Sections

@HWUSI-EAS611:34:6669YAAAXX:5:1:5069:1159 1:N:0:

- Starts with @ (required by fastq spec)
- Instrument ID (HWUSI-EAS611)
- Run number (34)
- Flowcell ID (6669YAAAXX)
- Lane (5)
- Tile (1)
- X-position (5069)
- Y-position (1159)
- [space]
- Read number (1)
- Was filtered (Y/N) (N) - You wouldn't normally see the Ys
- Control number (0 = no control)
- Sample number (only if demultiplexed using Illumina's software)
Phred Scores

• Start from (p) - the probability that the reported call is incorrect

• Initial transformation to a Phred score - positive integer from floating point

• Phred = \(-10 \times (\text{int})\log_{10}(p)\)
  
  • p=0.1  \hspace{1em} \text{Phred} = 10
  • p=0.01 \hspace{1em} \text{Phred} = 20
  • p=0.001 \hspace{1em} \text{Phred} = 30
Phred Score Encoding

- Translation of Phred score to single ASCII letter
- Based on standard ASCII table
- Can't translate directly
  - low values are non-printing
- Encode with Sanger System*
  - Phred+33

*Historically also had Illumina = Phred+64
Phred score encoding

: GBGGGGGGGGGDDGGDEDDGDGGGGGDDHDHGHGHBGG : G

: = ASCII 58

Phred33 encoding so Phred = 25

\[ p = 10^{\frac{25}{-10}} \]

\[ p = 0.003 \]
Phred score encoding

: GBGGGGGGGGGGDGDEDDGGGDDHDDHHGHGBGG

<table>
<thead>
<tr>
<th>Symbol</th>
<th>ASCII</th>
<th>Phred</th>
<th>Probability of miscall</th>
</tr>
</thead>
<tbody>
<tr>
<td>:</td>
<td>58</td>
<td>25</td>
<td>$p = 10^{(25/-10)} = 0.003$</td>
</tr>
<tr>
<td>G</td>
<td>71</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>

Better or Worse

<table>
<thead>
<tr>
<th>ASCII</th>
<th>Phred</th>
<th>Probability of miscall</th>
</tr>
</thead>
<tbody>
<tr>
<td>032{</td>
<td>4</td>
<td>072 H 092 \ 112 p</td>
</tr>
<tr>
<td>033 !</td>
<td>5</td>
<td>073 I 093 ] 113 q</td>
</tr>
<tr>
<td>034 &quot;</td>
<td>6</td>
<td>074 J 094 ^ 114 r</td>
</tr>
<tr>
<td>035 #</td>
<td>7</td>
<td>075 K 095 _ 115 s</td>
</tr>
<tr>
<td>036 $</td>
<td>8</td>
<td>076 L 096 ` 116 t</td>
</tr>
<tr>
<td>037 %</td>
<td>9</td>
<td>077 M 097 a 117 u</td>
</tr>
<tr>
<td>038 &amp;</td>
<td>0</td>
<td>078 N 098 b 118 v</td>
</tr>
<tr>
<td>039 '</td>
<td>59</td>
<td>079 O 099 c 119 w</td>
</tr>
<tr>
<td>040 (</td>
<td>060</td>
<td>080 P 100 d 120 x</td>
</tr>
<tr>
<td>041 )</td>
<td>061</td>
<td>081 Q 101 e 121 y</td>
</tr>
<tr>
<td>042 *</td>
<td>062</td>
<td>082 R 102 f 122 z</td>
</tr>
<tr>
<td>043 +</td>
<td>063</td>
<td>083 S 103 g 123 {</td>
</tr>
<tr>
<td>044 ,</td>
<td>064 @</td>
<td>084 T 104 h 124</td>
</tr>
<tr>
<td>045 -</td>
<td>065</td>
<td>085 U 105 i 125 ]</td>
</tr>
<tr>
<td>046 .</td>
<td>066 B</td>
<td>086 V 106 j 126 ~</td>
</tr>
<tr>
<td>047 /</td>
<td>067 C</td>
<td>087 W 107 k 127{ }</td>
</tr>
<tr>
<td>048 0</td>
<td>068 D</td>
<td>088 X 108 l 128 Ç</td>
</tr>
<tr>
<td>049 1</td>
<td>069 E</td>
<td>089 Y 109 m 129 ü</td>
</tr>
<tr>
<td>050 2</td>
<td>070 F</td>
<td>090 Z 110 n 130 é</td>
</tr>
<tr>
<td>051 3</td>
<td>071 G</td>
<td>091 [ 111 o 131 å</td>
</tr>
</tbody>
</table>
**Phred score encoding**

:GBGGGGGGGGGDGGDEDDGDGGDGDDGGHDHDHGHHHBGG:G

<table>
<thead>
<tr>
<th>Symbol</th>
<th>ASCII</th>
<th>Phred</th>
<th>Probability of miscall</th>
</tr>
</thead>
<tbody>
<tr>
<td>:</td>
<td>58</td>
<td>25</td>
<td>$p = 10^{(25/10)} = 0.003$</td>
</tr>
<tr>
<td>G</td>
<td>71</td>
<td>38</td>
<td>$p = 10^{(38/10)} = 0.00016$</td>
</tr>
</tbody>
</table>

```
032{  } 052 4 072 H 092 \ 112 p
033 ! 053 5 073 I 093 } 113 q
034 " 054 6 074 J 094 ^ 114 r
035 # 055 7 075 K 095 _ 115 s
036 $ 056 8 076 L 096 ` 116 t
037 % 057 9 077 M 097 a 117 u
038 & 058 0 078 N 098 b 118 v
039 ' 059 ; 079 O 099 c 119 w
040 ( 060 < 080 P 100 d 120 x
041 ) 061 = 081 Q 101 e 121 y
042 * 062 > 082 R 102 f 122 z
043 + 063 ? 083 S 103 g 123 { 
044 , 064 @ 084 T 104 h 124 |
045 - 065 A 085 U 105 i 125 }
046 . 066 B 086 V 106 j 126 ~
047 / 067 C 087 W 107 k 127{ }
048 0 068 D 088 X 108 l 128 Ç
049 1 069 E 089 Y 109 m 129 ü
050 2 070 F 090 Z 110 n 130 é
051 3 071 G 091 [ 111 o 131 â
```
# Phred score encoding

:GBGGGGGGGGGDGGDEDGDGGGGDHHDHGHHGBGG:GG

<table>
<thead>
<tr>
<th>Symbol</th>
<th>ASCII</th>
<th>Phred</th>
<th>Probability of miscall</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>71</td>
<td>38</td>
<td>$p = 10^{(38/-10)} = 0.00016$</td>
</tr>
<tr>
<td>B</td>
<td>66</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>

Better or worse

032 { } 052 4 072 H 092 \ 112 p
033 ! 053 5 073 I 093 ] 113 q
034 " 054 6 074 J 094 ^ 114 r
035 # 055 7 075 K 095 _ 115 s
036 $ 056 8 076 L 096 ` 116 t
037 % 057 9 077 M 097 a 117 u
038 & 058 ; 078 N 098 b 118 v
039 ' 059 ; 079 O 099 c 119 w
040 ( 060 < 080 P 100 d 120 x
041 ) 061 = 081 Q 101 e 121 y
042 * 062 > 082 R 102 f 122 z
043 + 063 ? 083 S 103 g 123 { 
044 , 064 @ 084 T 104 h 124 |
045 - 065 A 085 U 105 i 125 }
046 . 066 B 086 V 106 j 126 ~
047 / 067 C 087 W 107 k 127{ }
048 0 068 D 088 X 108 l 128 Ç
049 1 069 E 089 Y 109 m 129 ü
050 2 070 F 090 Z 110 n 130 é
051 3 071 G 091 [ 111 o 131 â
Phred score encoding

:GBGGGGGGGGGGGGDGGDEDDGDGGGGDHDDHGHHGHBGG:GG

<table>
<thead>
<tr>
<th>Symbol</th>
<th>ASCII</th>
<th>Phred</th>
<th>Probability of miscall</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>71</td>
<td>38</td>
<td>$p = 10^{38/10} = 0.00016$</td>
</tr>
<tr>
<td>B</td>
<td>66</td>
<td>33</td>
<td>$p = 10^{33/10} = 0.0005$</td>
</tr>
</tbody>
</table>

**WORSE**
Aligned Data – BAM files

• Expanded file containing alignment data as well as everything in the fastq file

• Two sections
  – Header (list of reference seqs and how the file was created)
  – Alignments (details of the alignments and sequences)

• Need special programs to read, normally 'samtools'
BAM Header

[andrewss@headstone Sample_lane1]$ samtools view -H lane1000_TTAGGC_test_L001_R1_GRCm38_hisat2.bam

@HD     VN:1.0  SO:unsorted
@SQ     SN:1   LN:195471971
@SQ     SN:10  LN:130694993
@SQ     SN:11  LN:122082543
@SQ     SN:12  LN:120129022
@SQ     SN:13  LN:120421639
...etc...

@PG     ID:hisat2   PN:hisat2   VN:2.1.0   CL:"/bi/apps/hisat2/2.1.0/hisat2-align-s --wrapper basic -0 --dta --sp 1000,1000 -p 7 -t --phred33-quals -x
/bi/scratch/Genomes/Mouse/GRCm38/Mus_musculus.GRCm38 --known-splicesite-infile
/bi/scratch/Genomes/Mouse/GRCm38/Mus_musculus.GRCm38.90.hisat2_splices.txt -U /tmp/17469.unp"
BAM Alignments

[andrewss@headstone Sample_lane1]$ samtools view
lane6045_TTAGGC_Col4_contr_L001_R1_trimmed_GRCm38_hisat2.bam | less

HWI-D00436:394:CBGLBANXX:1:1101:1222:1861 16  chr18 57944851 60 50M *
0 0 AAAAGATCTCTTGATTTAGAATTTTCTCTCAAAATGTGAGGGACTTTTATN
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGA<=># AS:i:-2 XN:i:0 XM:i:1 XO:i:0 XG:i:0
NM:i:1 MD:Z:49G0 YT:Z:UU NH:i:1

• Sections
  1. Sequence name
  2. Alignment Flags
  3. Reference sequence name
  4. Start position
  5. Mapping Quality (Phred)
  6. Alignment (CIGAR) string
  7. Paired sequence name
  8. Paired sequence position
  9. Total insert length
  10. Called Bases
  11. Base Quality String (Phred 33)
  12. Other Tags
BAM Flags

- A 12-bit binary number with a set of TRUE/FALSE values
  1. Sequence is paired end
  2. All reads from this template are aligned
  3. This read didn't align
  4. The paired read didn't align
  5. The read aligned in the reverse orientation
  6. The paired read aligned in the reverse orientation
  7. This is the first read
  8. This is the second read
  9. This is not the best alignment for this read
 10. This read failed upstream QC
 11. This read is a duplicate
 12. This is part of a chimeric alignment

https://broadinstitute.github.io/picard/explain-flags.html
What can QC tell us?
QC metrics can we work with

• Per Base Quality Scores

• Library composition
  – Base level
  – Sequence Level
  – Known sequences

• Mapping statistics

• Downstream Quantitation Values
Context is Key for QC

QC should be about what you expect and what you see
Context is Key for QC

Expectations

Individual Library:
- Universal
- Library Dependent

Replicate Libraries:
- Consistency
Some Software Packages for Sequence QC
FastQC

- Reads raw fastq files
- Performs multiple checks
  - Pass/warn/fail
  - Compares to genomic library
- HTML Report

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

- Reads fastq files
- Maps against a range of species / contaminants
- Identifies unexpected sequences in your library

https://www.bioinformatics.babraham.ac.uk/projects/fastq_screen/
Librarian

- Reads FastQ files
- Looks at the base composition
- Predict library type

Under development

https://github.com/DesmondWillowbrook/Librarian
MultiQC

- Aggregates QC information from multiple samples
- Large number of programs supported
- Combined HTML report

https://multiqc.info/
Assessing Universal Metrics
Universal QC Metrics

- Demultiplexing
- Base Call Quality
- Adapter Content
- Mapping Quality
Demultiplexing: Barcode Sequences

Barcodes shown explain 92% of the data

- GCCAATGT CW_F1S6
- TTAGGCAT CW_F1S3
- ACAATGCT CW_F1S5
- CGATGTCT CW_F1S2
- ATGACGTT CW_F1S1
- TGACCACG CW_F1S4
- GTATGCCG

Percentage of reads
Demultiplexing: Barcode Sequences

Barcodes shown explain 92% of the data

- TCTCGGTTCW_F1S14: 17%
- GGCTACAGCW_F1S11: 16%
- TCGTCTT: 15%
- TAAGCGTTCW_F1S15: 14%
- CAGATCTGCW_F1S7: 13%
- CTTGTACTCW_F1S12: 11%
- GATGCGC: 6%
- TCCGTCTTCWCW_F1S16: 2%
Demultiplexing: Barcode Sequences

Barcodes shown explain 91% of the data

- TCTCGGTT
- GGCTACAG
- TCCGTTT
- TAAGGGT
- CAGATCTG
- CTTGTACT
- AAGCGAGA CW_F1314
- CTGTAGCC CW_F1S11
- AAGACGGCA CW_F1S16
- AAGCGCTTA CW_F1S15
- CAGATCTG CW_F1S7
- ACTACAG CW_F1312

Percentage of reads
Demultiplexing: Barcode Sequences

Barcodes shown explain 99% of the data

Percentage of reads
Base Call Qualities – Per Cycle

Quality scores across all bases (Sanger / Illumina 1.9 encoding)

Phred Score

Read Position / Cycles of Chemistry
Base Call Qualities – Per Cycle

Quality scores across all bases (Sanger/Illumina 1.9 encoding)

Phred Score

Read Position / Cycles of Chemistry
Base Call Qualities – Per Cycle

Quality scores across all bases (Illumina 1.5 encoding)
Diagnosing Poor Base Call Qualities

• Not everything is bad

• Can see identify why some parts are bad and others aren't?

• May help to fix future runs
Per-Read Quality

- Are all reads equally affected?
- Is there a subset of reads which are always poor whilst others are good?
Positional Quality

Random Patterning

Position Specific Patterning

Tiles

Top

Bottom

Lanes

Per tile sequence quality
Read-through Adapters

![Diagram of Read-through Adapters with a 5' to 3' direction for Read 1 and Primer to Adapter.]
Measuring Read-through Adapters

![Graph showing % Adapter against position in read (bp) for different adapters.

- Primer
- Read 1
- Adapter 1
- Insert
- Adapter 2]
Clean-Up Options

Trimming 3’ end:
• Remove adapter read through
• Remove poor quality bases

Some quality issues may need to also remove specific reads

Despite issues may still be good enough for what is needed e.g. mapping
Mapping Statistics

Time loading forward index: 00:01:10
Time loading reference: 00:00:05
Multiseed full-index search: 00:20:47
24548251 reads; of these:
  24548251 (100.00%) were paired; of these:
    1472534 (6.00%) aligned concordantly 0 times
    21491188 (87.55%) aligned concordantly exactly 1 time
    1584529 (6.45%) aligned concordantly >1 times
94.00% overall alignment rate
Time searching: 00:20:52
Overall time: 00:22:02

If many reads do not map to the expected genome...
...Where do they come from?
Library Screening

Map your reads against a range of reference genomes

Classify matches as: unique to one species & single or multiple mapping
Library Screening

contaimated_screen

% Mapped

- Human
- Mouse
- Rat
- Ecoli
- Yeast
- PhiX
- Adapters
- Vectors
- Wasp
- rRNA
- No hits

Legend:
- One hit/one library
- Multiple hits/one library
- One hit/multiple libraries
- Multiple hits/multiple libraries
Library Screening

unguessable_screen

% Mapped

Human  Mouse  Rat  Ecoli  Yeast  PhiX  Adapters  Vectors  Wasp  rRNA  No hits

One hit\one library  Multiple hits\one library  One hit\multiple libraries  Multiple hits\multiple libraries
Assessing Library Dependent Metrics
Library Dependent QC Metrics

FastQC expects a Genomic Library: Do you?

Some QC metrics will be influenced by what you are sequencing

Concern or Expected?

- GC Content
- Base Composition
- Duplication
Library GC Content

- Generic summary of library composition at a read level
- Expect a normally distributed set of values centred on the overall GC content
Sharp Peaks in GC
Concern or Expected

Specific Contamination with single sequence or closely related sequences
Artificial sequences, ribosomal RNA, contaminants
Broader Peaks in GC
Concern or Expected

More extensive mixture of reads with different GC content
GC Skew
Concern or Expected

More extensive subset of reads with extreme differences in GC
For every chemistry cycle we can look at the number of ATGC we call.
For Libraries with random start positions the composition should be the same for all cycles.
Bias Composition Throughout
Concern or Expected

Wrong Sequence

Amplicon

Proportional biases of bases at specific positions: Very low diversity
Bias Composition Throughout
Concern or Expected

WGBS

Consistent disproportional expression of bases
Bias Composition at 5’ end
Concern or Expected

ATAC – Transposases

RRBS – Restriction Start Site

Proportional biases of bases at the start of a read: A preferred start site
Bias Composition at 3’ end
Concern or Expected

Proportional biases of bases at the end of a read: consistent closing sequence
Bias Composition at 3’ end
Concern or Expected

Proportional biases of bases at the end of a read: consistent closing sequence

Bioinformatics processing can also influence QC metrics!
• How frequently the exact same sequence appears in your library
• For WGS expect most sequences to be unique
Duplication

If the exact same sequence appears more than once it could be...

Technical:

- ATCCGAGCTATTCCGGCGAGCTCGCCAGTTACG
- ATCCGAGCTATTCCGGCGAGCTCGCCAGTTACG
- ATCCGAGCTATTCCGGCGAGCTCGCCAGTTACG

Coincidental:

- ATCCGAGCTATTCCGGCGAGCTCGCCAGTTACG
- ATCCGAGCTATTCCGGCGAGCTCGCCAGTTACG
- ATCCGAGCTATTCCGGCGAGCTCGCCAGTTACG

- PCR duplicates
- Deep sequencing
- Highly present sequences
- Restricted diversity libraries
Duplication
Concern or Expected

RNA-Seq

Amplicon

BUT could have technical duplication with expected coincidental duplication!
Overrepresented Sequences

- Extreme duplication
- The exact same sequence is a significant proportion of the whole library (which might not be duplicated overall)

  - Poly Sequences
  - Specific Sequences
Sources of Poly Sequences

PolyN – Quality too poor to make any calls

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>AA</td>
<td>462344</td>
<td>1.07009784553387</td>
</tr>
<tr>
<td>GG</td>
<td>232548</td>
<td>0.5382147642627897</td>
</tr>
<tr>
<td>AA</td>
<td>127291</td>
<td>0.2946155309984244</td>
</tr>
<tr>
<td>GG</td>
<td>87792</td>
<td>0.2032949571694588</td>
</tr>
<tr>
<td>AA</td>
<td>85181</td>
<td>0.19715176672683803</td>
</tr>
<tr>
<td>GG</td>
<td>48918</td>
<td>0.1132209753507845</td>
</tr>
</tbody>
</table>

PolyG – Empty space in 2 colour chemistry, can be technically “high quality” calls

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
</table>
| GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
Sources of Poly Sequences

PolyA (or PolyT) – Common in RNA-Seq

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT</td>
<td>68355</td>
<td>1.7344041279604823</td>
</tr>
<tr>
<td>AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA</td>
<td>67792</td>
<td>1.7201186595230343</td>
</tr>
</tbody>
</table>
Overrepresented Specific Sequences

- Normally artificial sequences (primers, adapters, vectors etc)
- Can search a database of known sequences to find matches

Example of an Adapter dimer:
Overrepresented Specific Sequences

- Other potential sources...

**Ribosomal**

**Contaminant**
Overrepresented Specific Sequences

Which of these libraries would you expect to flag with overrepresented specific sequences?

A) Whole Genome Bisulfite Library
B) Amplicon Library
C) RNAseq Library
Overrepresented Specific Sequences

Which of these libraries would you expect to flag with overrepresented sequences?

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Count</th>
<th>Percentage</th>
<th>Possible Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGGACGGAGGGCTATTTCGATTTTTCATATTTGCATATACGATA</td>
<td>70194</td>
<td>37.84369541308145</td>
<td>No Hit</td>
</tr>
<tr>
<td>GGGACGGAGGGCTATTTCGATTTTTCATATTTGCATATACGATA</td>
<td>55496</td>
<td>29.91956175087446</td>
<td>No Hit</td>
</tr>
<tr>
<td>GATGAGAGGGCTATTTCGATTTTTCATATTTGCATATACGATA</td>
<td>40221</td>
<td>21.684350132625994</td>
<td>No Hit</td>
</tr>
<tr>
<td>TGGACGGAGGGCTATTTCGATTTTTCATATTTGCATATACGATA</td>
<td>281</td>
<td>0.1514955467857066</td>
<td>No Hit</td>
</tr>
</tbody>
</table>
Assessing Consistency
Aggregated Statistics

Individual QC reports are useful but helpful to have a wider picture

MultiQC currently has modules to support 128 different bioinformatics tools, https://multiqc.info/modules/
6 Samples, 24 files:
Paired end Pre/post trimming
Aggregated Mapping Stats

Identify local QC problems by spotting samples that behave differently
Putting QC into Practice
Expectations and Observations are Key

Can you tell if these libraries are any good?
QC In A Nut-Shell

Expectations → Observations → Critical Analysis → Understanding & Confidence

Good Science 😊
QCFAIL.com

Articles about common next-generation sequencing problems
Exercise: Assess the Quality of Public Sequencing Datasets

Consider Expectations

- RNA-Seq
- ChIP-Seq
- WGBS
- ATAC-Seq

Fastq File

QC Metrics:
- FastQC

Trimmed Fastq File

QC Metrics:
- FastQC
- FastQ Screen

Aligned File e.g. bam file

QC Metrics:
- Aligner programme dependent QC stats
- e.g. mapping statistics

Review Observations

- FastQC
- Bismark
- FastScreen
- Bowtie
- Hisat2
- MultiQC