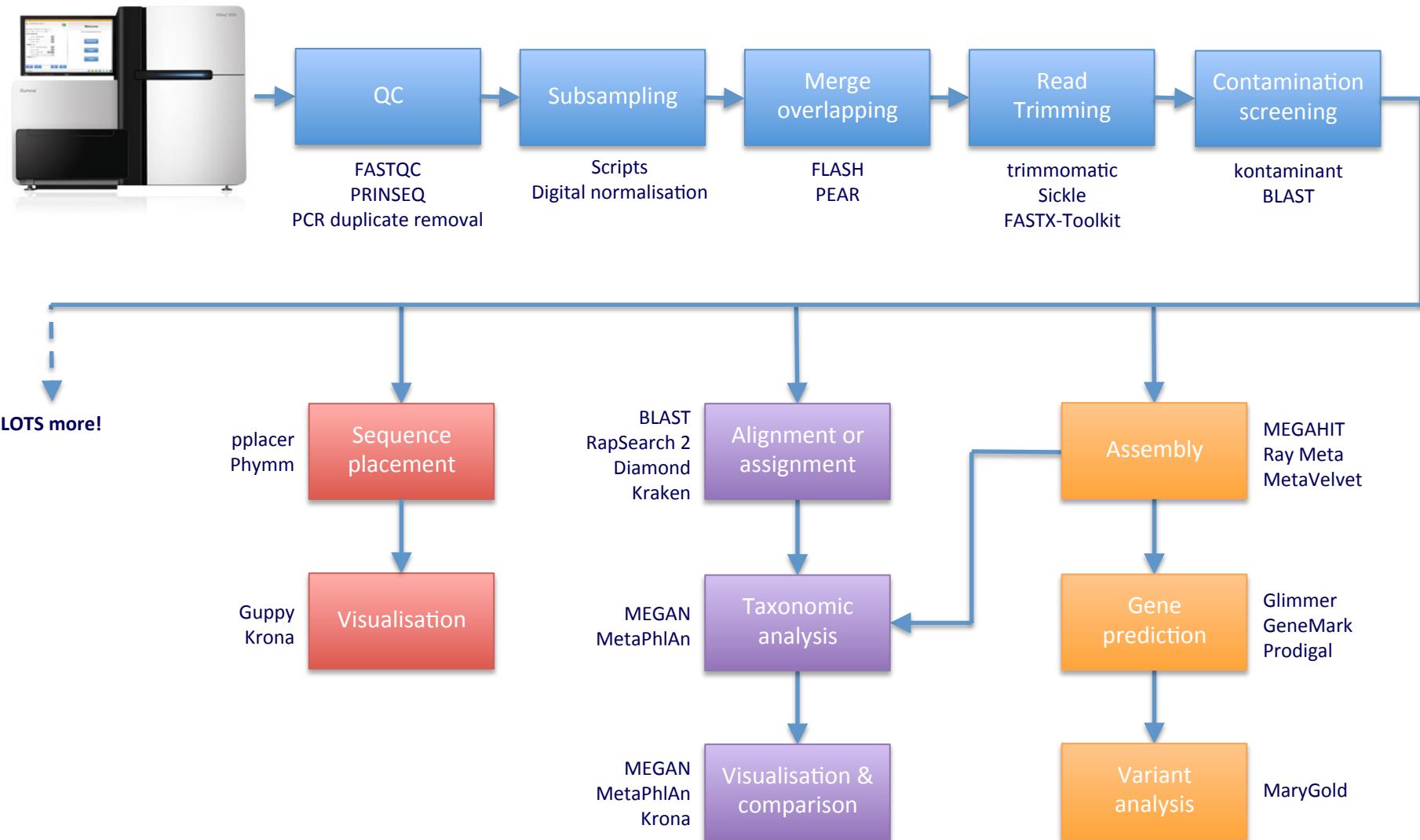


Next generation sequencing data analysis for metagenomics

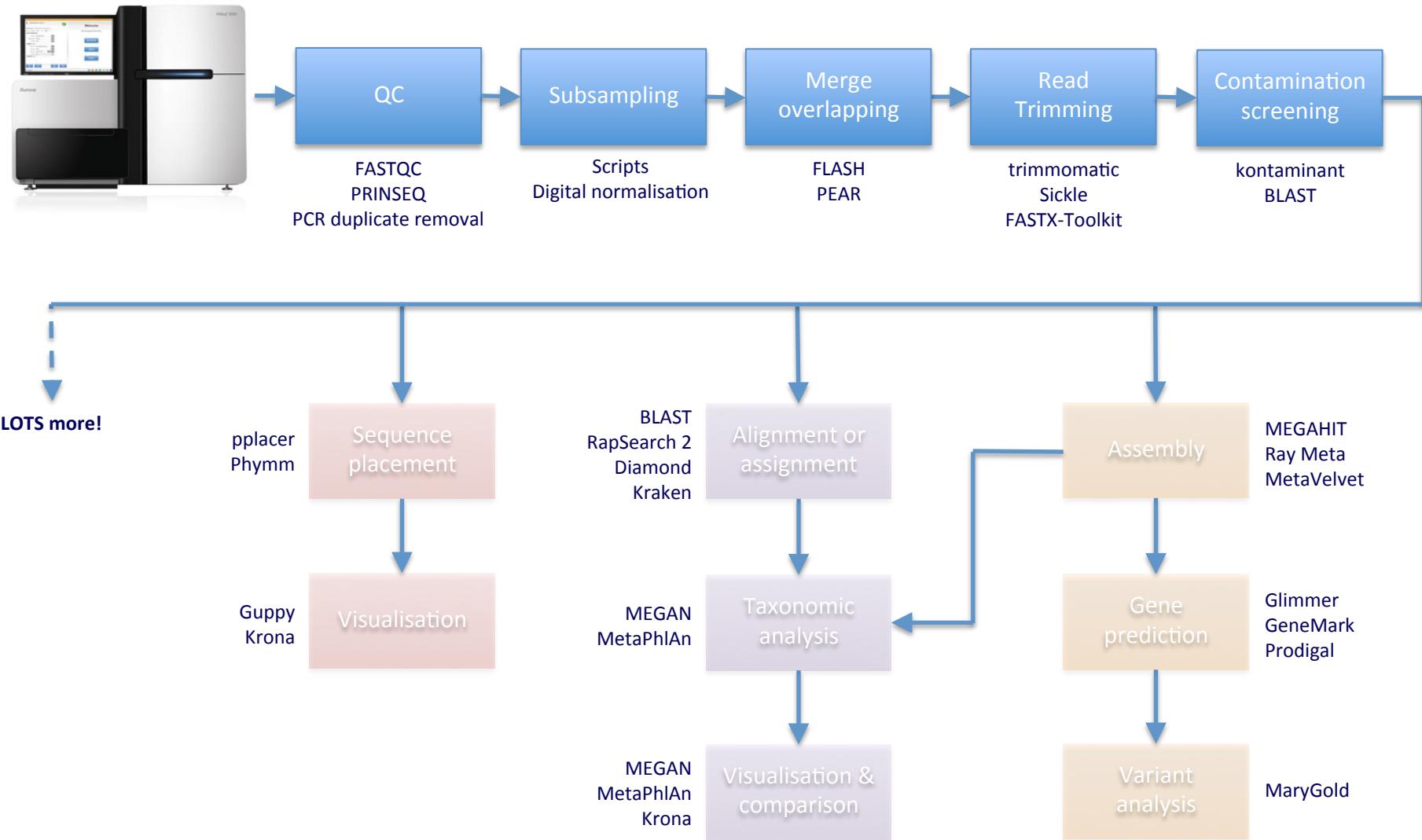
Richard Leggett

Richard.Leggett@tgac.ac.uk
@richardmleggett

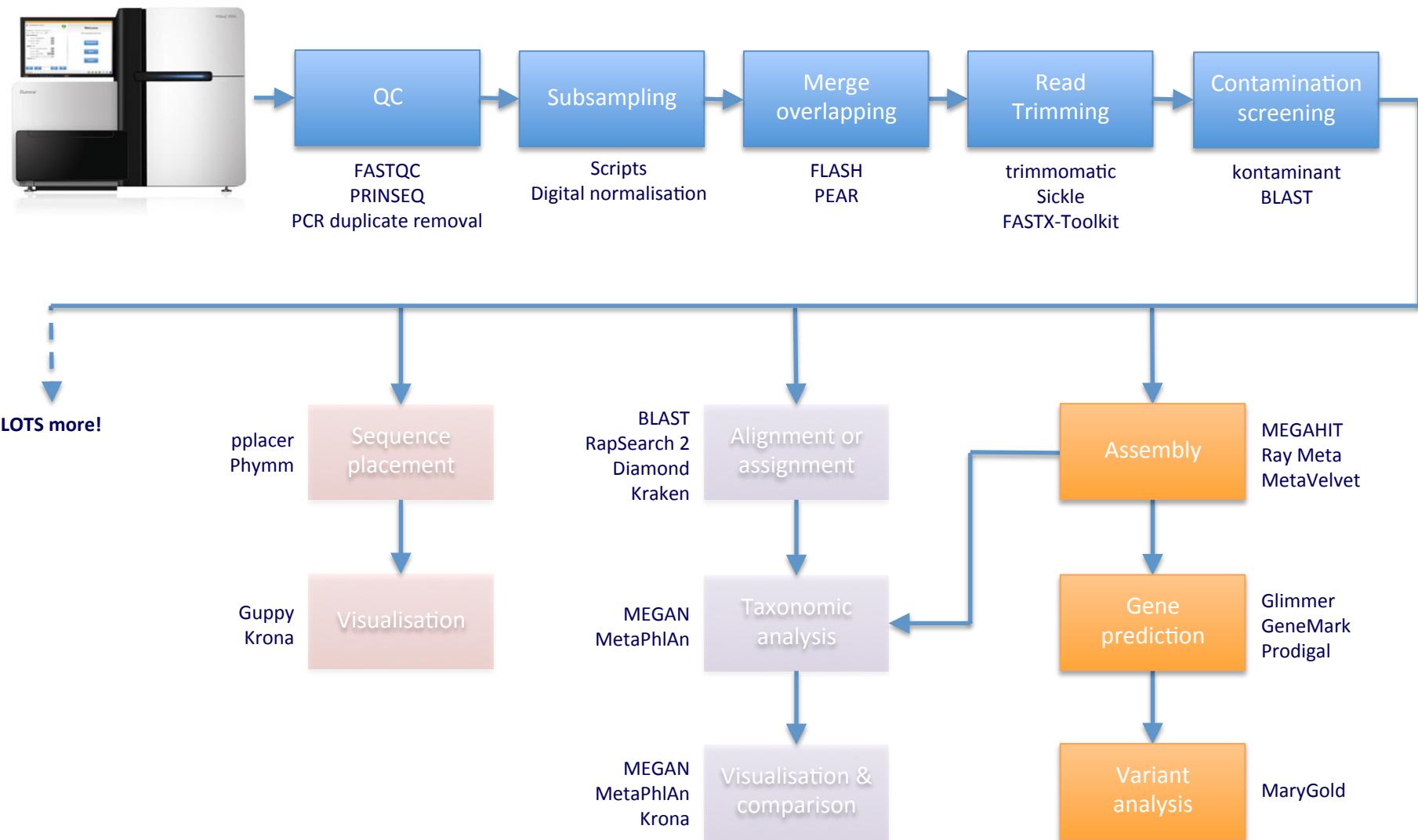
Typical pipelines



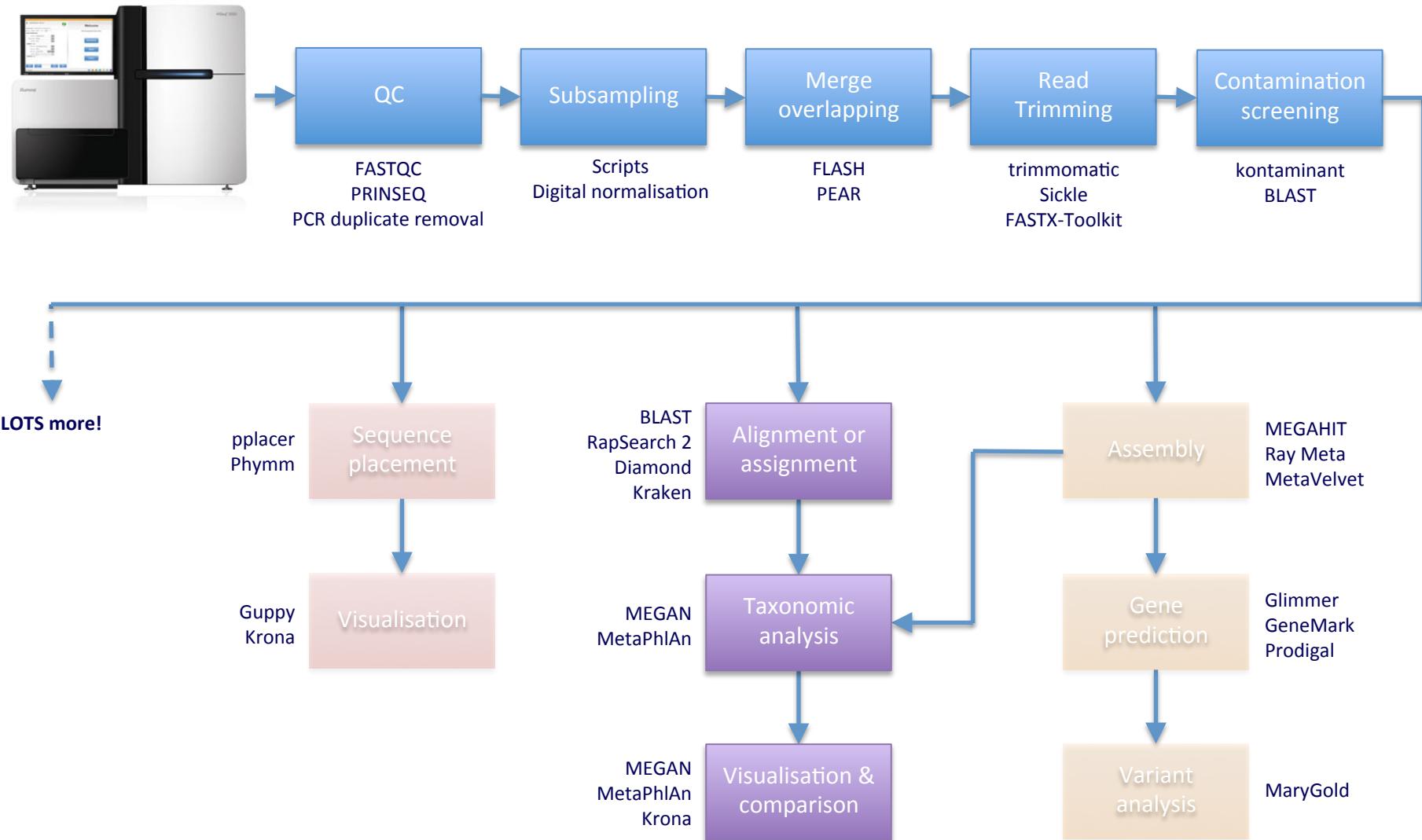
Typical pipelines



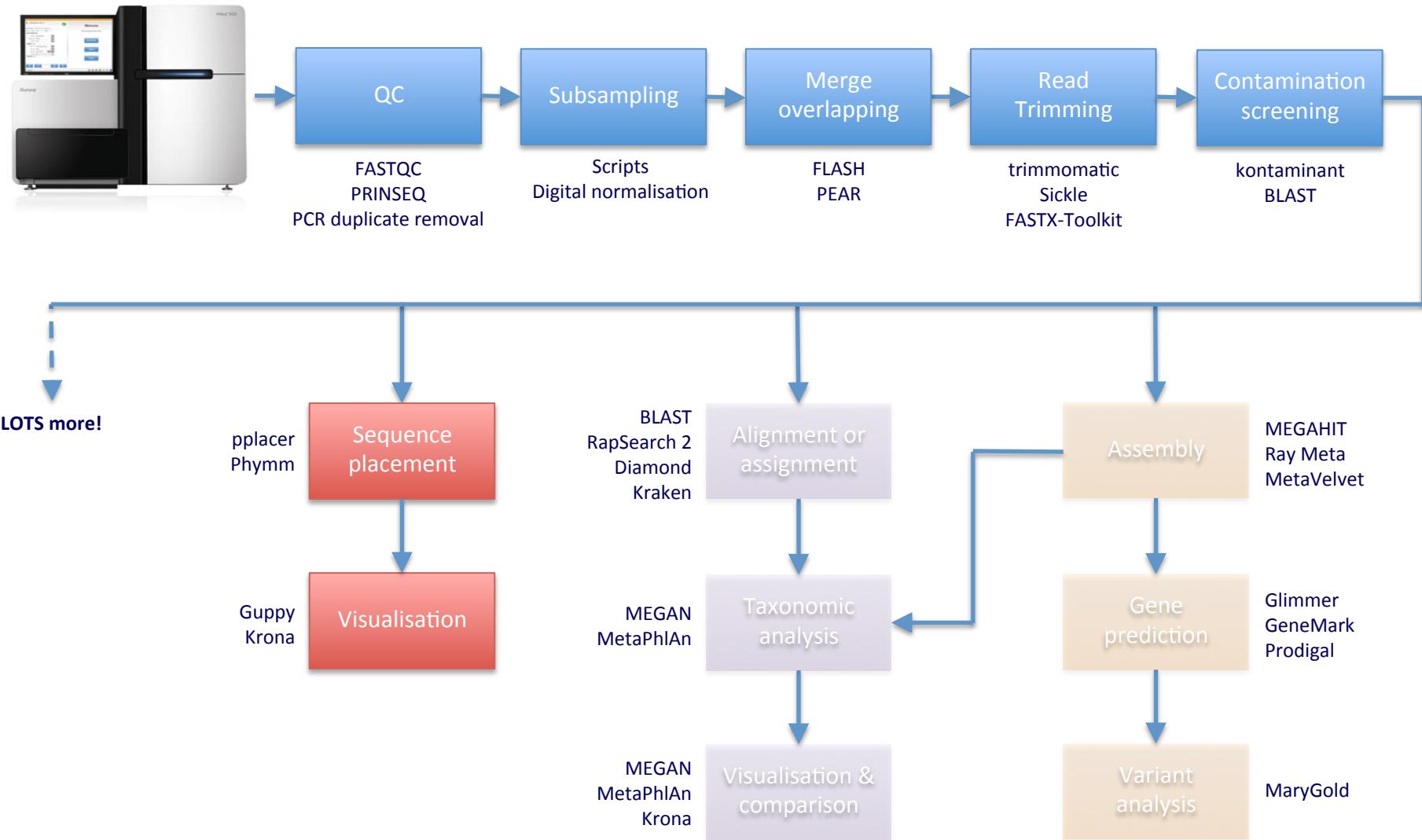
Typical pipelines



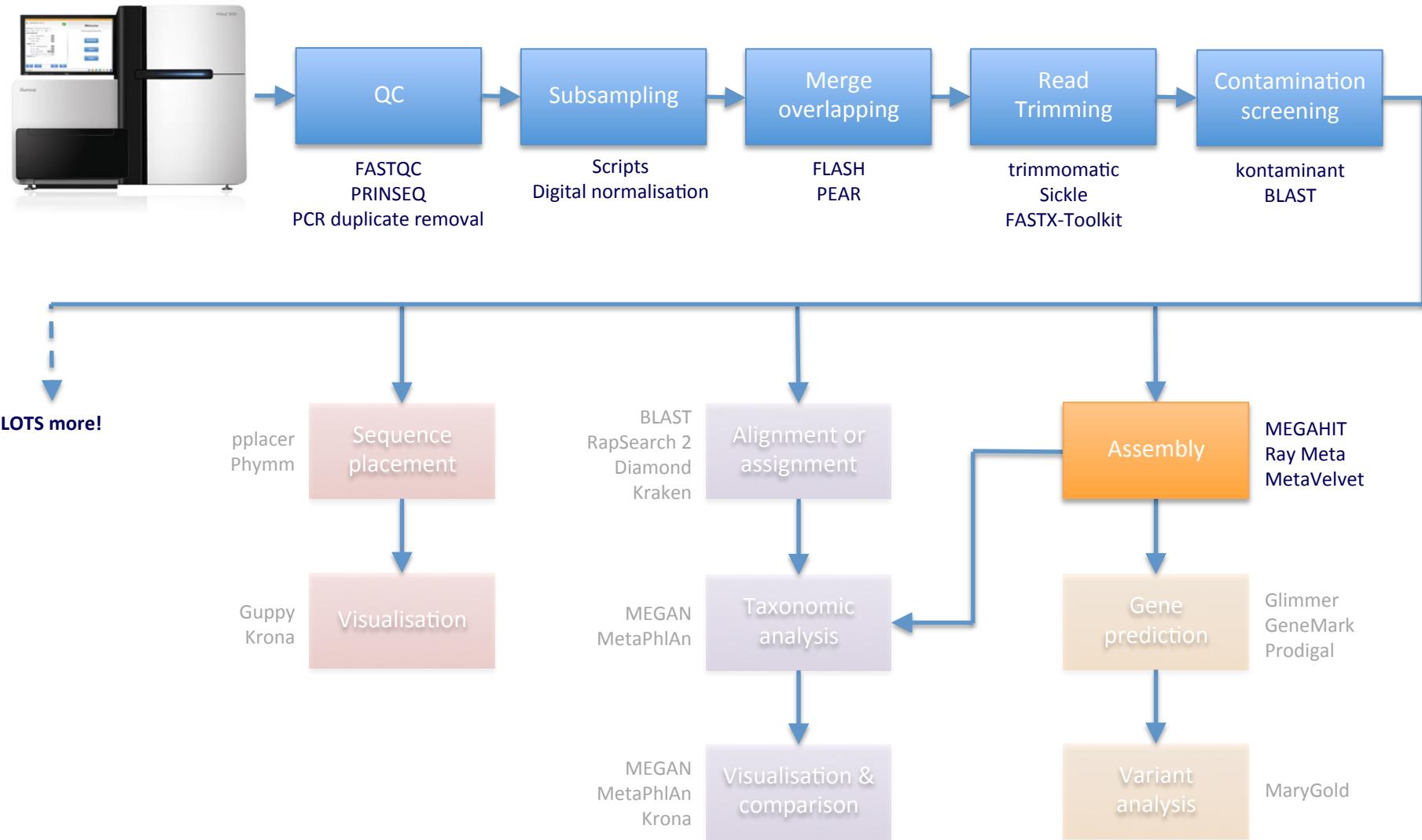
Typical pipelines



Typical pipelines



In this session...



Firstly... sequence data...



```
@HWI-ST790:234:D0W8BACXX:1:1101:1792:2000 1:N:0:GCCAA
ACNATTAACAACCTGGTGTTCAGCATGAGAACTATCTGCAGCTGAGTCTCGTATCCGTGACG
+
CC#4ADDFHIIHHHIIIEGHIIIIIIIIIGIIIIIIIIIIIIIDGH HIDHHIII6@FGI
@HWI-ST790:234:D0W8BACXX:1:1101:2592:1999 1:N:0:GCCAA
CTNGAATGCAGGTAGAATACATCTCCGGATAAGCCTCGCGGCCCGGGGGGGGGAGAG
+
:=#44AA?:<DFFE>FED?3A<EHH>FIF?ADGCGBA?D#####
@HWI-ST790:234:D0W8BACXX:1:1101:4221:1999 1:N:0:GCCAA
GGNAATACGAAAGATAAGCTACGCAAGAACGAAGGATTACTGCGAAAGGCTGCGATGCGGCA
+
@#4=BDDFDFHDIIBGIHHIGGIIIBHHIF=ABB@?B<DE@BF<FHH@@EHACD<B3=8@:B
```

FASTQ files

- e.g. Illumina read files
- 4 lines per read
- Stores sequence and quality information

```
Read ID → @HWI-ST790:234:D0W8BACXX:1:1101:1792:2000 1:N:0:GCCAA
Sequence → ACNATTAACAACCTGGTGGTCAGCATGAGAACATTATCTGCAGCTGAGTCTCGTATCCGTGACG
+
Quality → CC#4ADDFHBBBBHIIIEGHIIIIIIIIIGIIIIIIIIIIIDGHIDHHIII6@FGI
@HWI-ST790:234:D0W8BACXX:1:1101:2592:1999 1:N:0:GCCAA
CTNGAATGCAGGTAGAATAACATCTCCGGATAAGCCTCGCGCCCCGGGGGGGGAGAG
+
:=#44AA?:<DFFE>FED?3A<EHH>FIF?ADGCGBA?D#####:######
@HWI-ST790:234:D0W8BACXX:1:1101:4221:1999 1:N:0:GCCAA
GGNAAATACGAAAGATAAGCTACGCAAGAACGAAGGATTACTGCGAAAGGCTGCGATGCGGCA
+
@@#4=BDDDFDFHDIIBGIHHHIGGIIIBHHIF=ABB@?B<DE@BF<FHH@EHACD<B3=8@:B
```

FASTQ files

- Sanger format quality scores 0-93
- Encoded with ASCII characters 33-126
- Older versions of Illumina software slightly different

Quality value representations														
Char	ASCII	Q	Char	ASCII	Q	Char	ASCII	Q	Char	ASCII	Q	Char	ASCII	Q
!	33	0	4	52	19	G	71	38	Z	90	57	m	109	76
"	34	1	5	53	20	H	72	39	[91	58	n	110	77
#	35	2	6	54	21	I	73	40	\	92	59	o	111	78
\$	36	3	7	55	22	J	74	41]	93	60	p	112	79
%	37	4	8	56	23	K	75	42	^	94	61	q	113	80
&	38	5	9	57	24	L	76	43	-	95	62	r	114	81
'	39	6	:	58	25	M	77	44	`	96	63	s	115	82
(40	7	;	59	26	N	78	45	a	97	64	t	116	83
)	41	8	<	60	27	O	79	46	b	98	65	u	117	84
*	42	9	=	61	28	P	80	47	c	99	66	v	118	85
+	43	10	>	62	29	Q	81	48	d	100	67	w	119	86
,	44	11	?	63	30	R	82	49	e	101	68	x	120	87
-	45	12	@	64	31	S	83	50	f	102	69	y	121	88
.	46	13	A	65	32	T	84	51	g	103	70	z	122	89
/	47	14	B	66	33	U	85	52	h	104	71	{	123	90
0	48	15	C	67	34	V	86	53	i	105	72		124	91
1	49	16	D	68	35	W	87	54	j	106	73	}	125	92
2	50	17	E	69	36	X	88	55	k	107	74	~	126	93
3	51	18	F	70	37	Y	89	56	l	108	75			

FASTQ files

- Q score relates to probability, p, that base is incorrect:

$$Q_{\text{sanger}} = -10 \log_{10} p$$

- What this means...

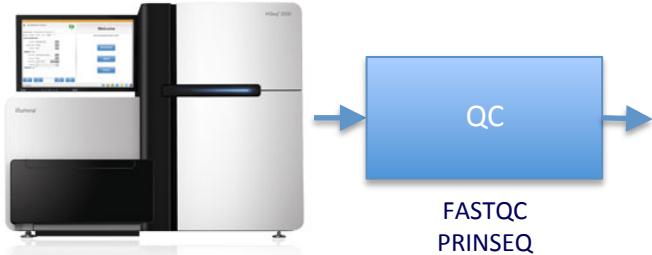
Phred Quality Score	Probability of Incorrect Base Call	Base Call Accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1,000	99.9%
40	1 in 10,000	99.99%
50	1 in 100,000	99.999%

FASTA files

- e.g. assembler contigs
- Stores ID and sequence data only
- Sequence data can cover multiple lines

Sequence ID → >contig1
Sequence → ACNATTAACAACCTTGGTGTTCAGCATGAGAACCTTATCTGCAGCTGAGTCTCGTATCCGTGACG
CTGAGTCTCGTATCCGTGACGGTTAGGGCGATTAGCATAGA
>contig2
TGACTAGCGGATT CGATT CGGAGG CTTAT GGGCATT CCAGAT GCAGCT AGCAGAT GACATAGAT
GGGCATT
>contig3
CCCCCTGACTAGCGGATT CGGTT CAGCATGAGTACGAATT CGGAGG CTTAT GGGCATT CCAGA
AGCGTGCAGCTAGCAGATGAAGCGCATAGATGGGCTATT GTTCAGCATGAGCTGATCAACTACG
TACGGGACTGAGATGCCATGCAGTTGG
>contig4
TGACTAGCTAGTGGATTGACGAC
...

Quality Control

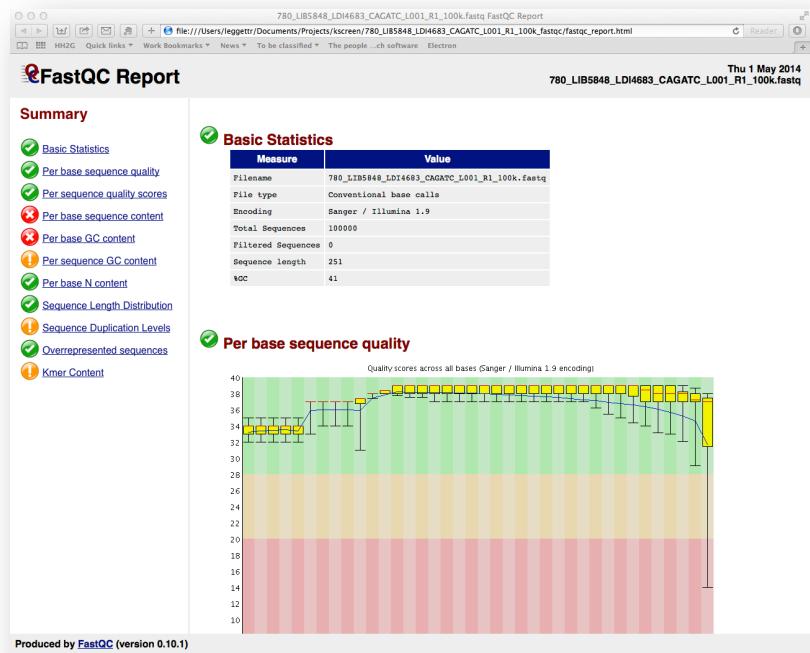


Why?

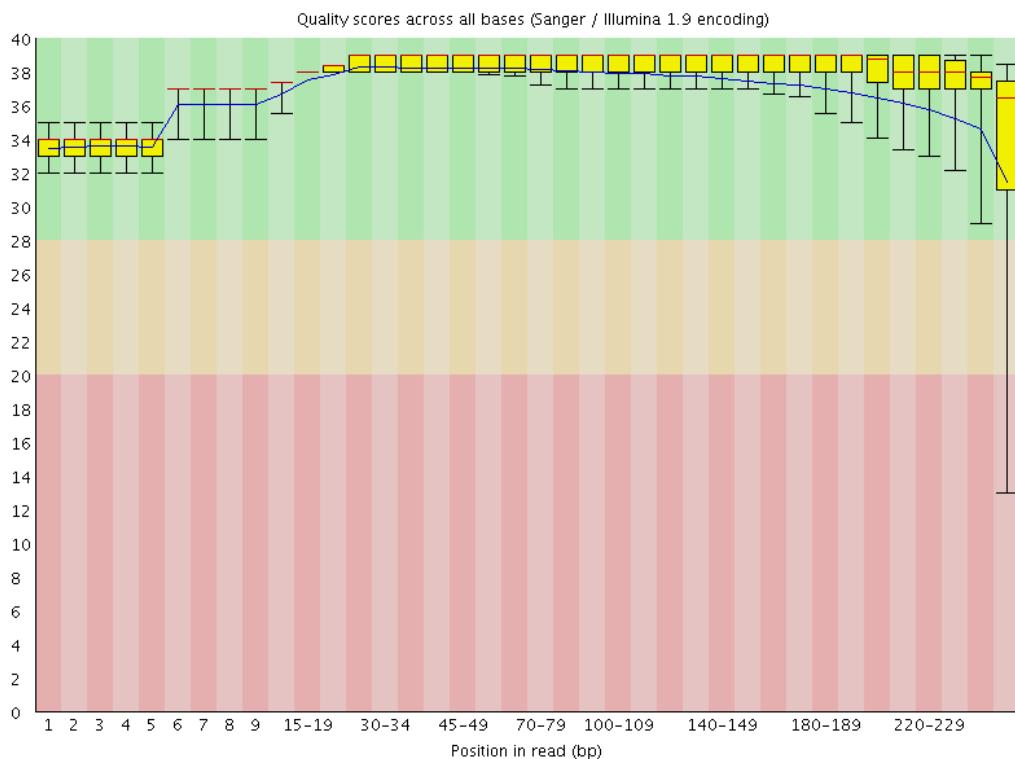
- Look for problems at an early stage.
- Understand if sequencing worked.
- Know your data = better results!

FASTQC

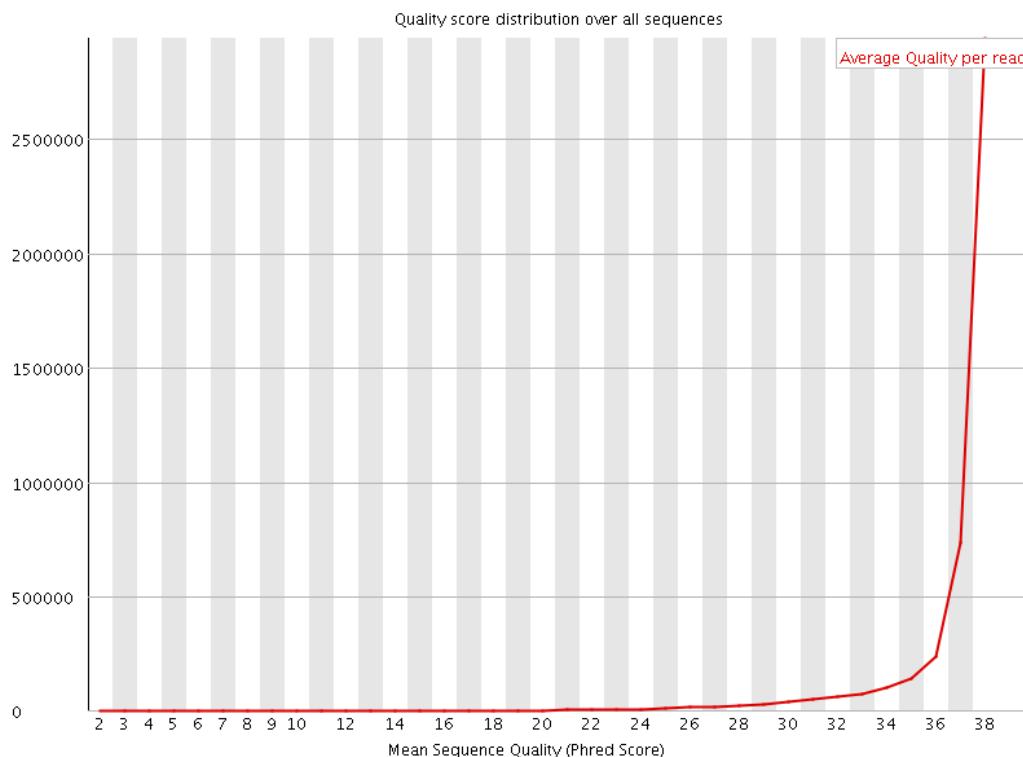
- Developed by Simon Andrews, Babraham Institute
- Quality control checks on sequence data
 - Quality scores
 - GC content
 - Duplicated sequence
 - Adaptors
 - Over-represented sequence
 - and more...
 - Run interactive or CLI



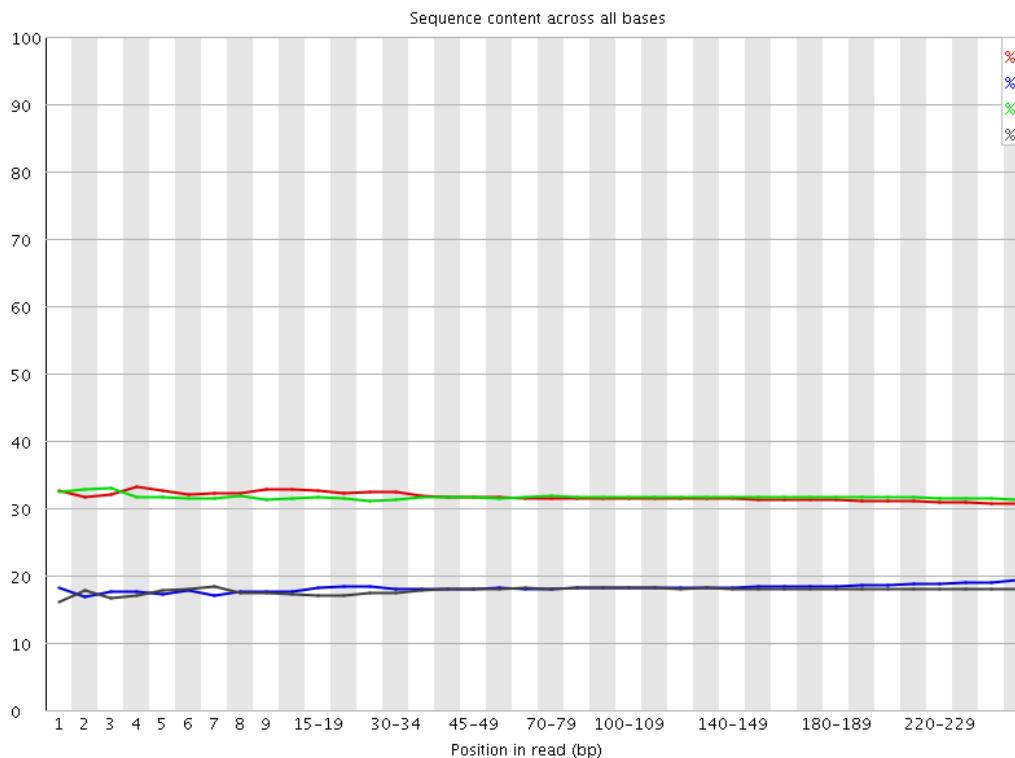
- Per-base quality
- “Average quality at position x”



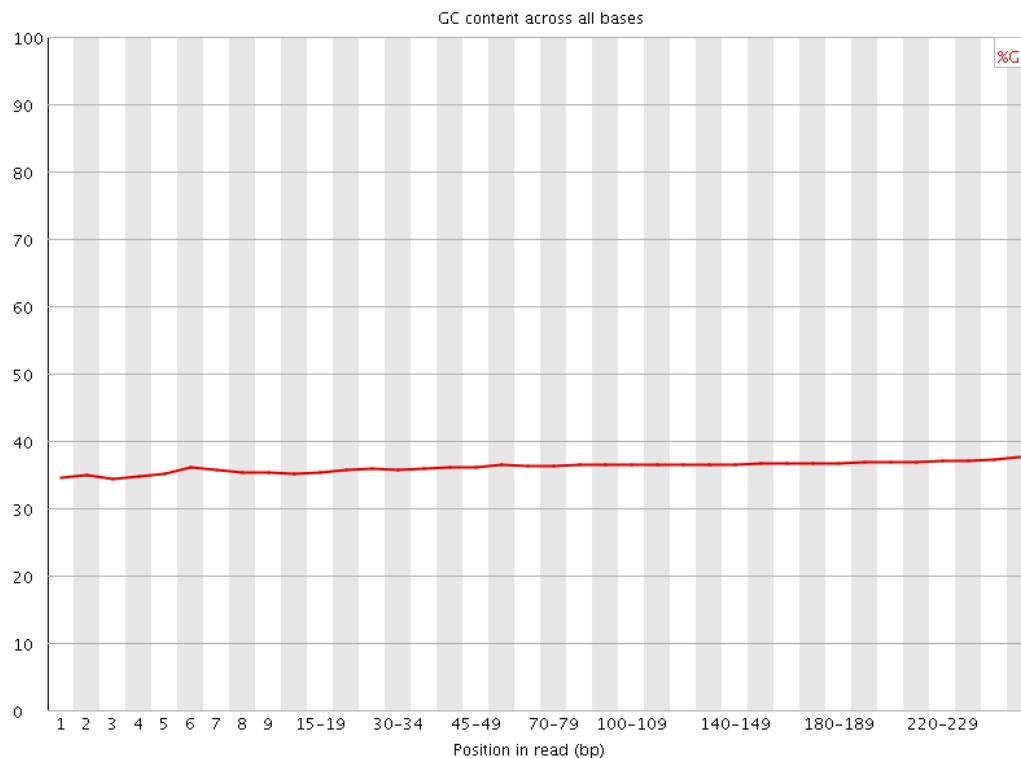
- Per-sequence quality
- “Number of reads with average quality x”



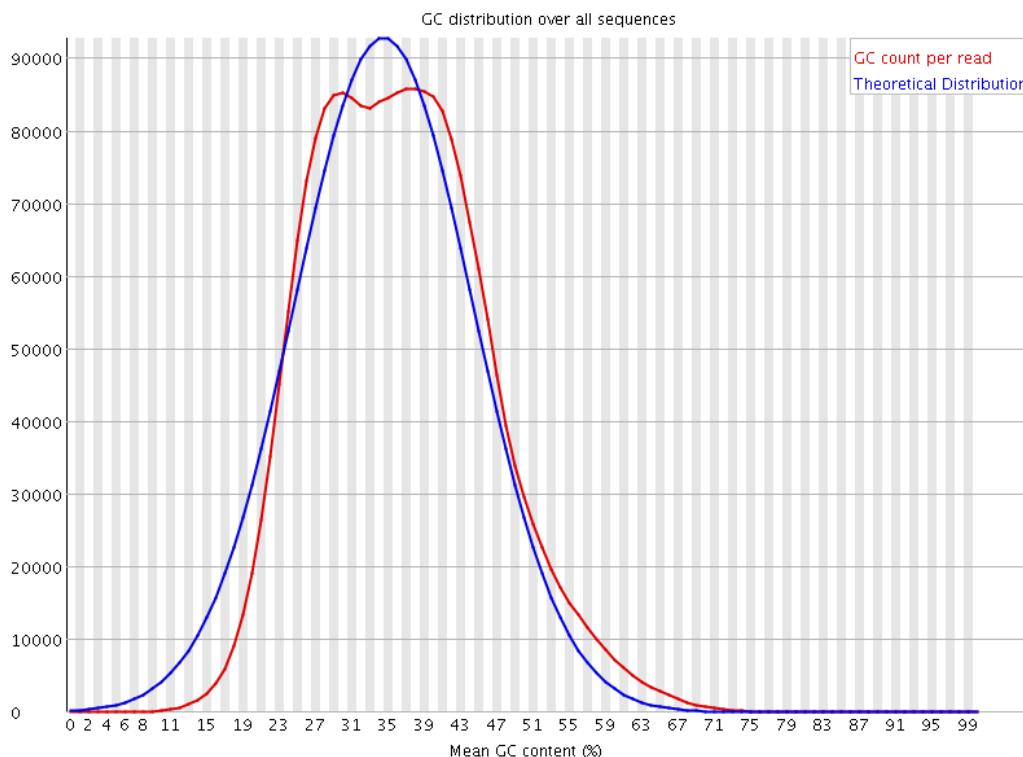
- Per-base sequence content
- “% of reads with A, C, G, T at position x”



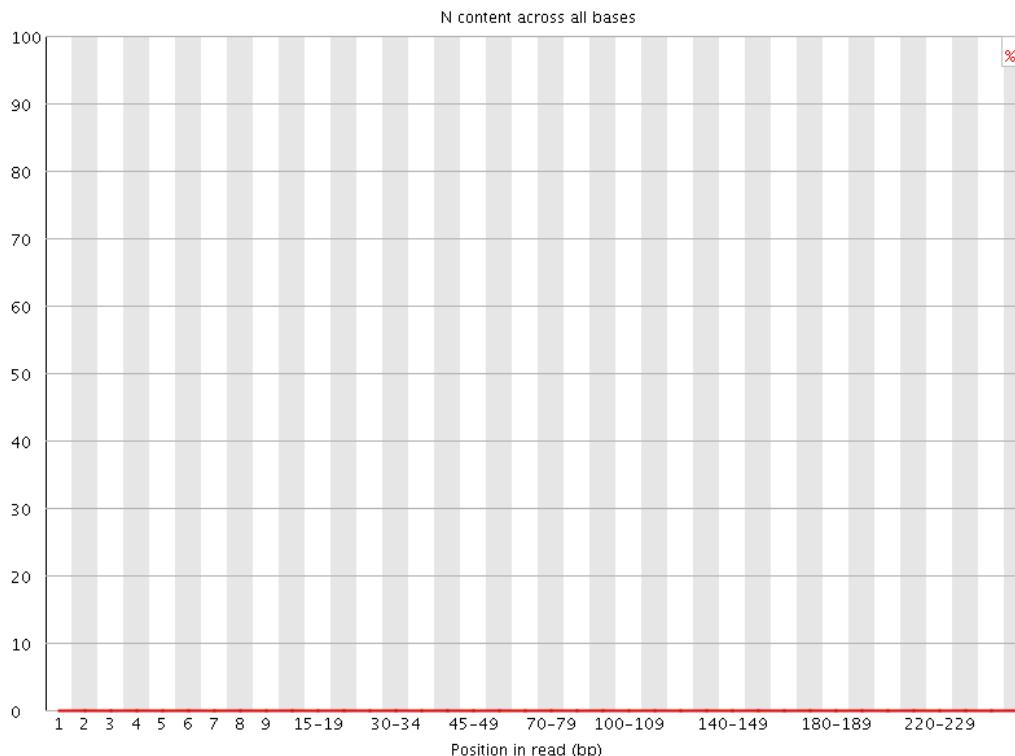
- Per-base GC content
- “Average GC % at position x across all reads”



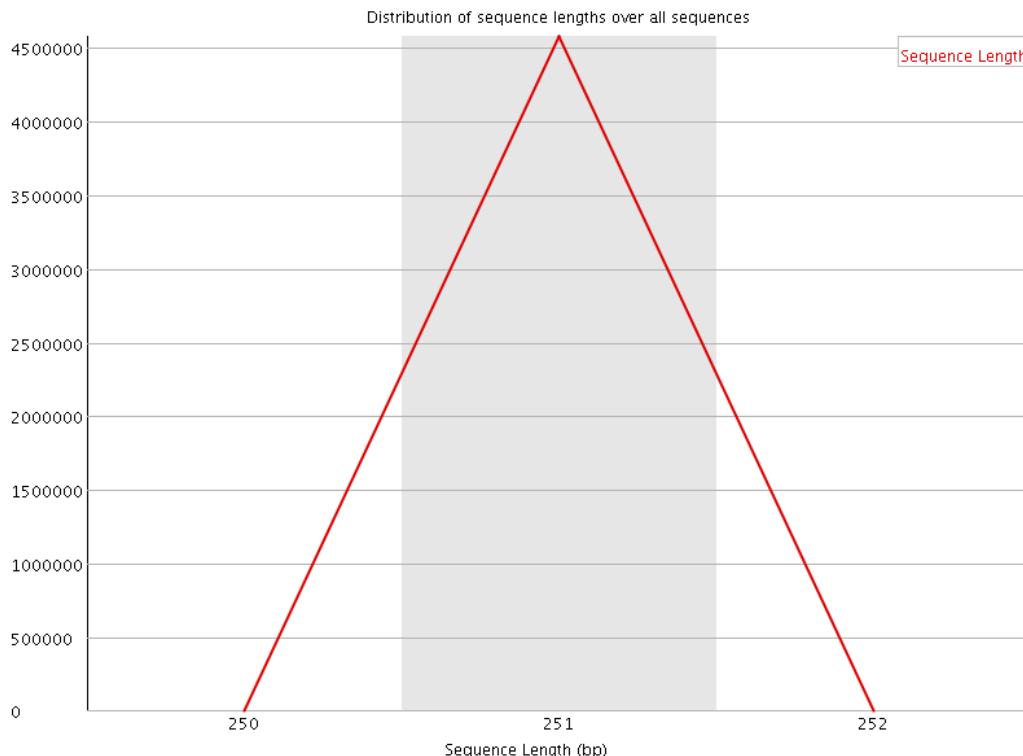
- Per-sequence GC content
- “Number of reads with GC % x”



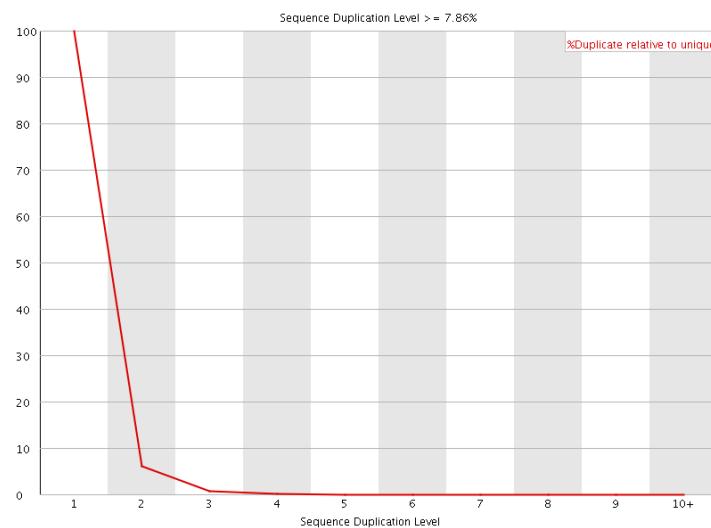
- Per-base N content
- “Percentage of Ns at position x”



- Sequence length distribution
- “Number of reads of length x”



- Duplication levels
- “Relative number of sequences with duplication x”
- Shown relative to sequences that appear once.
- First 50 bases used.



...but will not remove duplicates for you...

PCR duplicate removal

- Can reduce analysis overhead and remove bias.
- Options:

- FASTX Toolkit e.g.

```
fastx_collapse -i input.fastq -j output.fastq
```

Can be slow, single end only.

- SamTools rmdup e.g.

```
samtools rmdup input.srt.bam output.bam
```

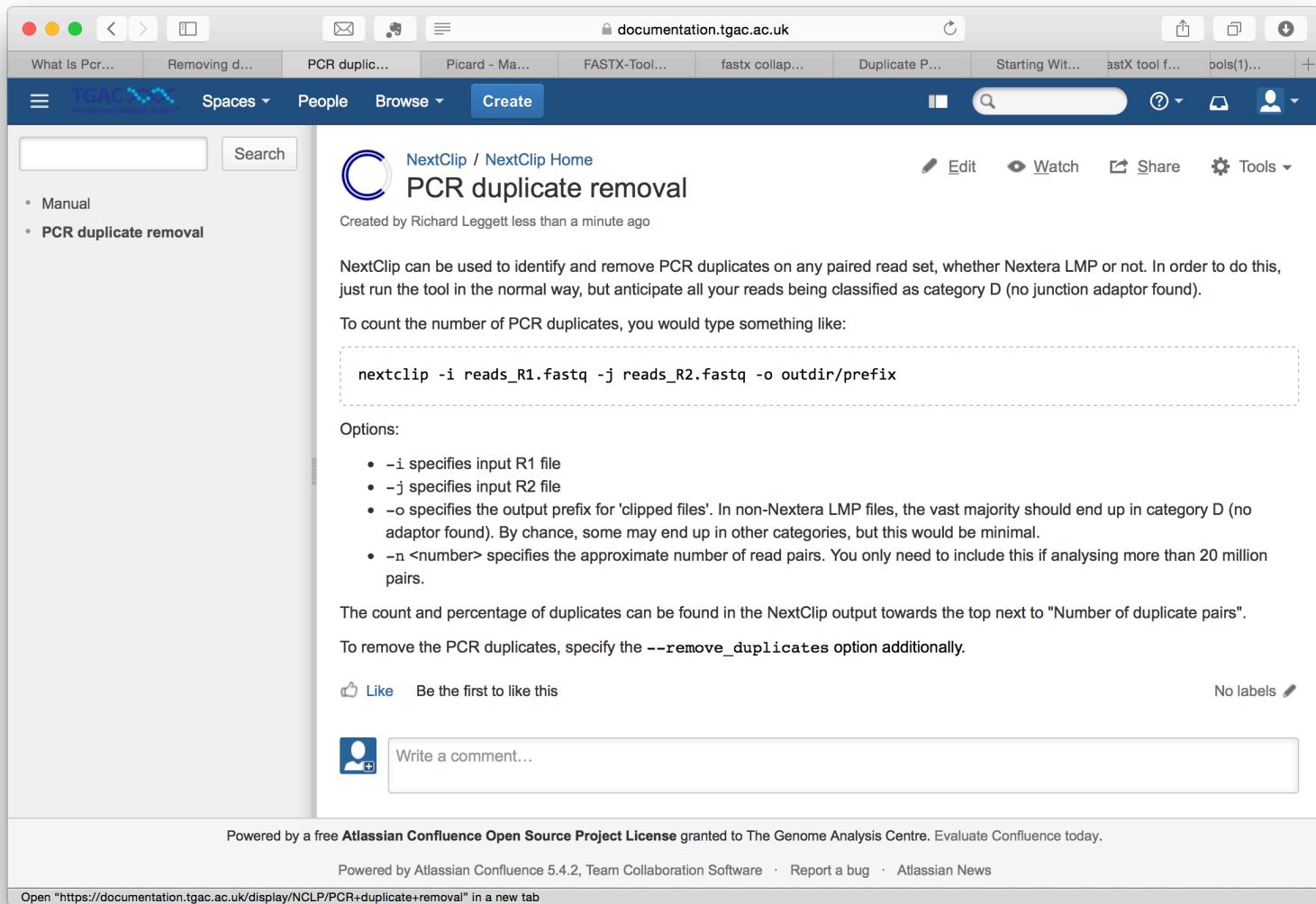
Requires alignment, e.g. with BWA first.

- NextClip – can use just for PCR analysis/removal

- Pair-end only
 - To run...

```
nextclip -i reads_R1.fastq -j reads_R2.fastq  
-o outdir/prefix
```

- <https://documentation.tgac.ac.uk/>



The screenshot shows a web browser window for the URL <https://documentation.tgac.ac.uk/>. The page title is "PCR duplicate removal". The page content includes a brief description of NextClip's function, a command-line example, options for input and output files, and instructions for removing duplicates. At the bottom, there are social sharing icons and a comment section.

NextClip / NextClip Home

PCR duplicate removal

Created by Richard Leggett less than a minute ago

NextClip can be used to identify and remove PCR duplicates on any paired read set, whether Nextera LMP or not. In order to do this, just run the tool in the normal way, but anticipate all your reads being classified as category D (no junction adaptor found).

To count the number of PCR duplicates, you would type something like:

```
nextclip -i reads_R1.fastq -j reads_R2.fastq -o outdir/prefix
```

Options:

- `-i` specifies input R1 file
- `-j` specifies input R2 file
- `-o` specifies the output prefix for 'clipped files'. In non-Nextera LMP files, the vast majority should end up in category D (no adaptor found). By chance, some may end up in other categories, but this would be minimal.
- `-n <number>` specifies the approximate number of read pairs. You only need to include this if analysing more than 20 million pairs.

The count and percentage of duplicates can be found in the NextClip output towards the top next to "Number of duplicate pairs".

To remove the PCR duplicates, specify the `--remove_duplicates` option additionally.

Like Be the first to like this No labels

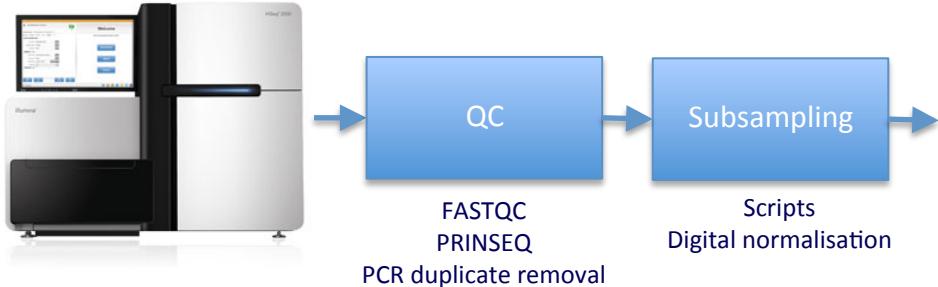
Write a comment...

Powered by a free Atlassian Confluence Open Source Project License granted to The Genome Analysis Centre. Evaluate Confluence today.

Powered by Atlassian Confluence 5.4.2, Team Collaboration Software · Report a bug · Atlassian News

Open "<https://documentation.tgac.ac.uk/display/NCLP/PCR+duplicate+removal>" in a new tab

Subsampling



Why?

- Quick look at data.
- Protein BLAST can be SLOW.
- You can have too much coverage in assembly.

Subsampling

- How?
 - People tend to do their own thing!
 - Mainly “quick” scripts – Google it.
 - Could take top X reads of a file...
 - ...but much better to random sample.
 - Also, consider *digital normalisation* for assembly.

Random read sampling

- <https://www.biostars.org/p/6544/>

Question: Selecting Random Pairs From Fastq?



I'd like to select a random subset of Illumina PE reads from a set of files. I realize this is pretty trivial to do, but I thought I'd ask if there's a simple program available to do so? It's important that it retains both ends of each pair, and in the right order, and that it can work by streaming over the input sequences (obviously they don't fit in RAM).

9

Or do I just have to write it myself?



illumina fastq sequence random code • 15k views



4.6 years ago by
Ketil • 3.6k
Germany

[ADD COMMENT](#)

• [link](#) • Not following ▾

modified 14 months ago by [Alex Reynolds](#) • 12k • written 4.6 years ago by [Ketil](#) • 3.6k



Not sure I follow - are you saying a random sampling might introduce bias that wasn't there already? This seems to go against the definition of 'random'.

2

[ADD REPLY](#) • [link](#)

written 4.6 years ago by [Ketil](#) • 3.6k



Do you expect your coverage to be even, otherwise you might get in hot water by introducing biases when randomly sampling over the whole genome/region: <http://biostar.stackexchange.com/questions/4340/downsampling-bam-files>

[ADD REPLY](#)

• [link](#)

written 4.6 years ago by [Allpowerde](#) • 1.1k



In one command line:

14

```
paste f1.fastq f2.fastq | \ #merge the two fastqs
awk '{ printf("%s",$0); n++; if(n%4==0) { printf("\n");} else { printf("\t\t");}'
}' | \ #merge by group of 4 lines
shuf | \ #shuffle
head | \ #only 10 records
sed 's/\t\t/\n/g' | \ #restore the delimiters
awk '{print $1 > "file1.fastq"; print $2 > "file2.fatsq"}' #split in two files.
```



4.6 years ago by
Pierre Lindenbaum • 75k
France

Random read sampling

- My own Perl script available:

<https://github.com/richardmleggett/scripts>

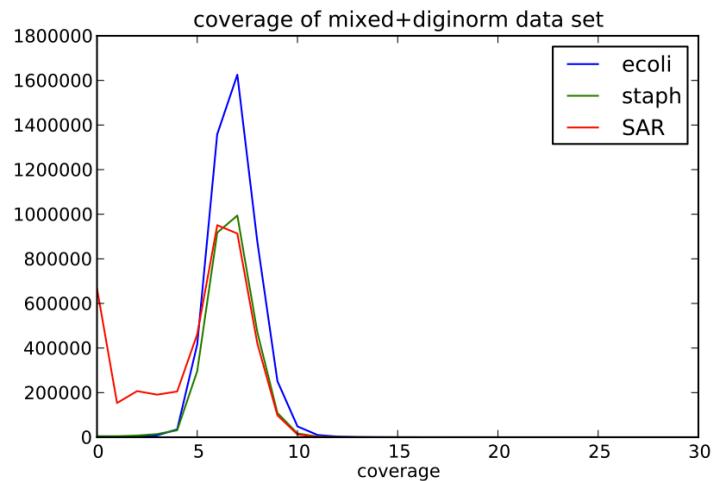
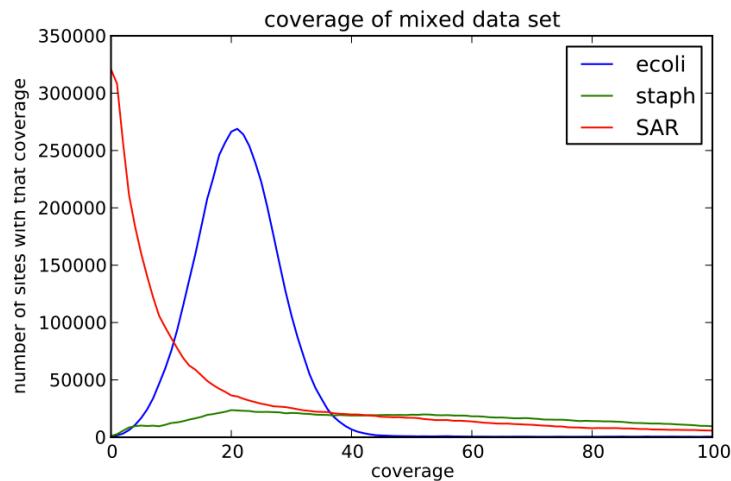
- To run:

```
subsample.pl -fastq -a R1in.fastq -b R2in.fastq  
             -c R1out.fastq -d R2out.fastq
```

- Optionally, also get the “remainder” of reads using -e and -f options.

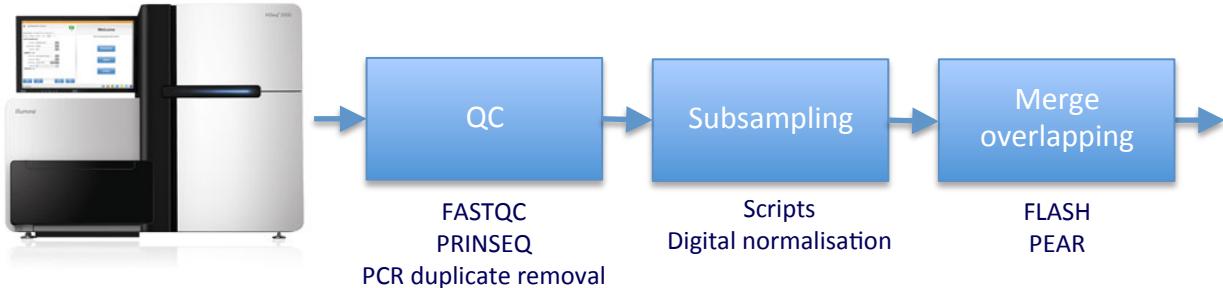
Digital normalisation

- “Abundance normalization” for assembly.



- <http://ivory.idyll.org/blog/what-is-diginorm.html>
- Implemented in khmer tool. Paper:
 - <http://f1000research.com/articles/4-900/>

Merging overlapping reads



Why?

- Overlapping Paired Ends.
- Longer matches.
- Better assemblies.

- Tunable mismatch ratio and minimum overlap.
- Outputs merged reads, plus files of unmerged.
- Simple to run:

```
flash R1.fastq R2.fastq -o output_prefix
```

BIOINFORMATICS

ORIGINAL PAPER

Vol. 27 no. 21 2011, pages 2957–2963
doi:10.1093/bioinformatics/btr507

Genome analysis

Advance Access publication September 7, 2011

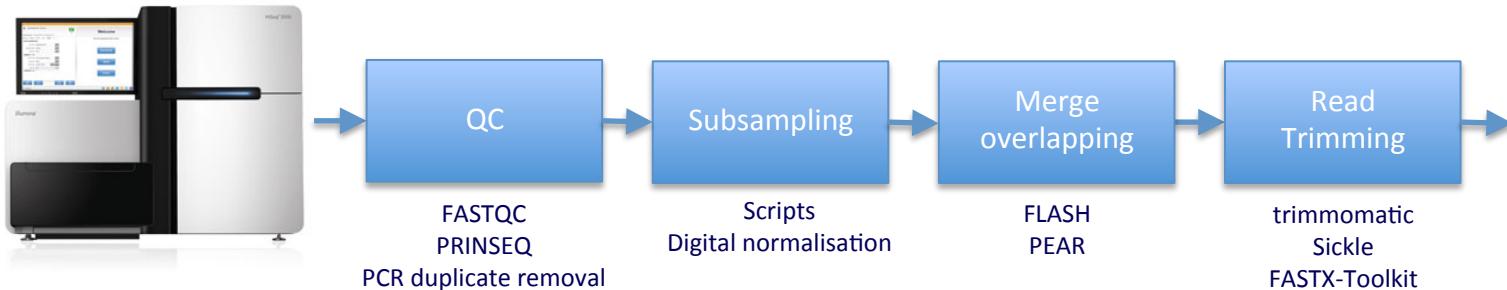
FLASH: fast length adjustment of short reads to improve genome assemblies

Tanja Magoč* and Steven L. Salzberg

McKusick-Nathans Institute of Genetic Medicine, Johns Hopkins University School of Medicine, Baltimore,
MD 21205, USA

Associate Editor: Martin Bishop

Read trimming



Why?

- Remove adaptor contamination.
- Remove poor quality bases.
- Get better matching/assembly.

Read trimming

- May include:
 - Quality clipping – removing low quality bases from end
 - Ns – remove reads containing Ns
 - Adaptor trimming – removing primer/adaptor contamination
- Lots of tools:
 - Sickle – windowed quality trimming
 - Trimmomatic – adaptor and quality trimming
 - FASTX-Toolkit – adaptor and quality trimming
 - Cutadapt – adaptor trimming
 - and many more...

Read trimming

- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read: TGACGGCTATCGATCGATTAGCGGATAGCCATGACGCAAT
Qualities: 12345779:; >ACDFGHIIIIIGFFDCA?< : 97533210

Read trimming

- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read: TGACGGCTATCGATCGATTAGCGGATAGCCATGACGCAAT
Qualities: 12345779:; >ACDFGHIIIIIGFFDCA?< : 97533210



Mean quality 17.5

Read trimming

- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read: TGACGGCTATCGATCGATTAGCGGATAGCCATGACGCAAT

Qualities: 12345779:; >ACDFGHIIIIIGFFDCA?< : 97533210



Mean quality 18.5

Read trimming

- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read: TGACGGCTATCGATCGATTAGCGGATAGCCATGACGCAAT
Qualities: 123456779:; >ACDFGHIIIIIGFFDCA?< : 97533210



Mean quality 19.5

Read trimming

- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read: TGACGGCTATCGATCGATTAGCGGATAGCCATGACGCAAT

Qualities: 12345779:; >ACDFGHIIIIIGFFDCA?< : 97533210



Mean quality 20.75

Read trimming

- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read: TGAC**CGGC**TATCGATCGATTAGCGGATAGCCATGACGCAAT
Qualities: 123**45779**:; >ACDFGHIIIIIGFFDCA?< : 97533210



Mean quality 20.75

Read trimming

- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read: TGACGGCTATCGATCGATTAGCGGATAGCCATGACGCAAT
Qualities: 12345779:; >ACDFGHIIIIIGFFDCA?< : 97533210

Read trimming

- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read: TGACGGCTATCGATCGATTAGCGGATAGCCATGACGCAAT
Qualities: 12345779:; >ACDFGHIIIIIGFFDCA?< : 97533210



Mean quality 19.5

Read trimming

- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read: TGACGGCTATCGATCGATTAGCGGATAGCCATGACGGAAAT
Qualities: 12345779:; >ACDFGHIIIIIGFFDCA?< : 97533210

Mean quality 19.5

Read trimming

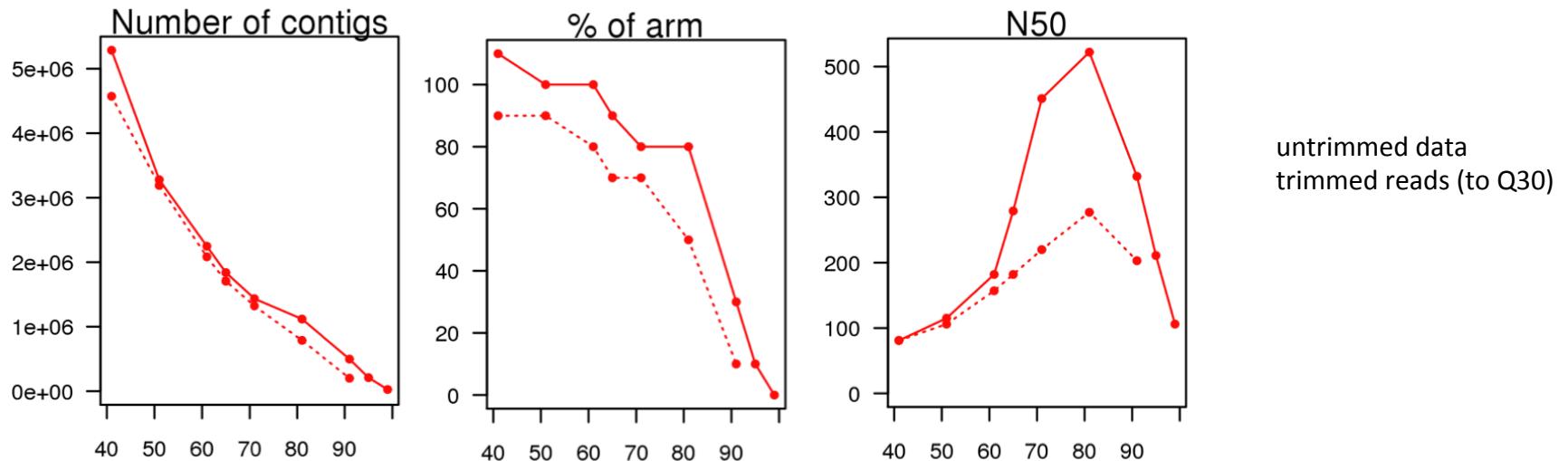
- Sliding windows
 - Sickle uses window size $0.1 \times$ read length
 - Example: Read length 40, window size 4, quality threshold 20

Read:	TGAC	GCAAT
Qualities:	1234	33210

Trimmed read: GGCTATCGATCGATTAGCGGATAGCCATGAC
5779:; >ACDFGHIIIIIGFFDCA?< : 975

Read trimming

- Quality trimming can have a cost – eg. hexaploid wheat



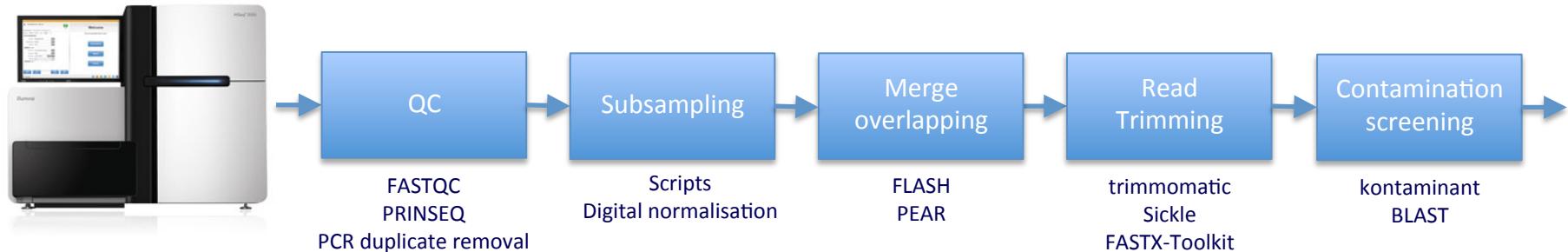
- Gene content of trimmed assembly was lower
- Indicates cost to quality trimming
- Good quality sequence lost as well as low quality

(Data from Paul Bailey, Bernardo Clavijo at TGAC)

Read trimming

- Adaptor trimming approaches (eg. Trimmomatic):
 - Database of adaptors and primers
 - Matching bases increase score
 - Mismatches reduce score
 - Good enough match triggers clipping
- Most tools also allow
 - Hard clipping at start or end
 - Minimum read size specification

Contamination screening



Why?

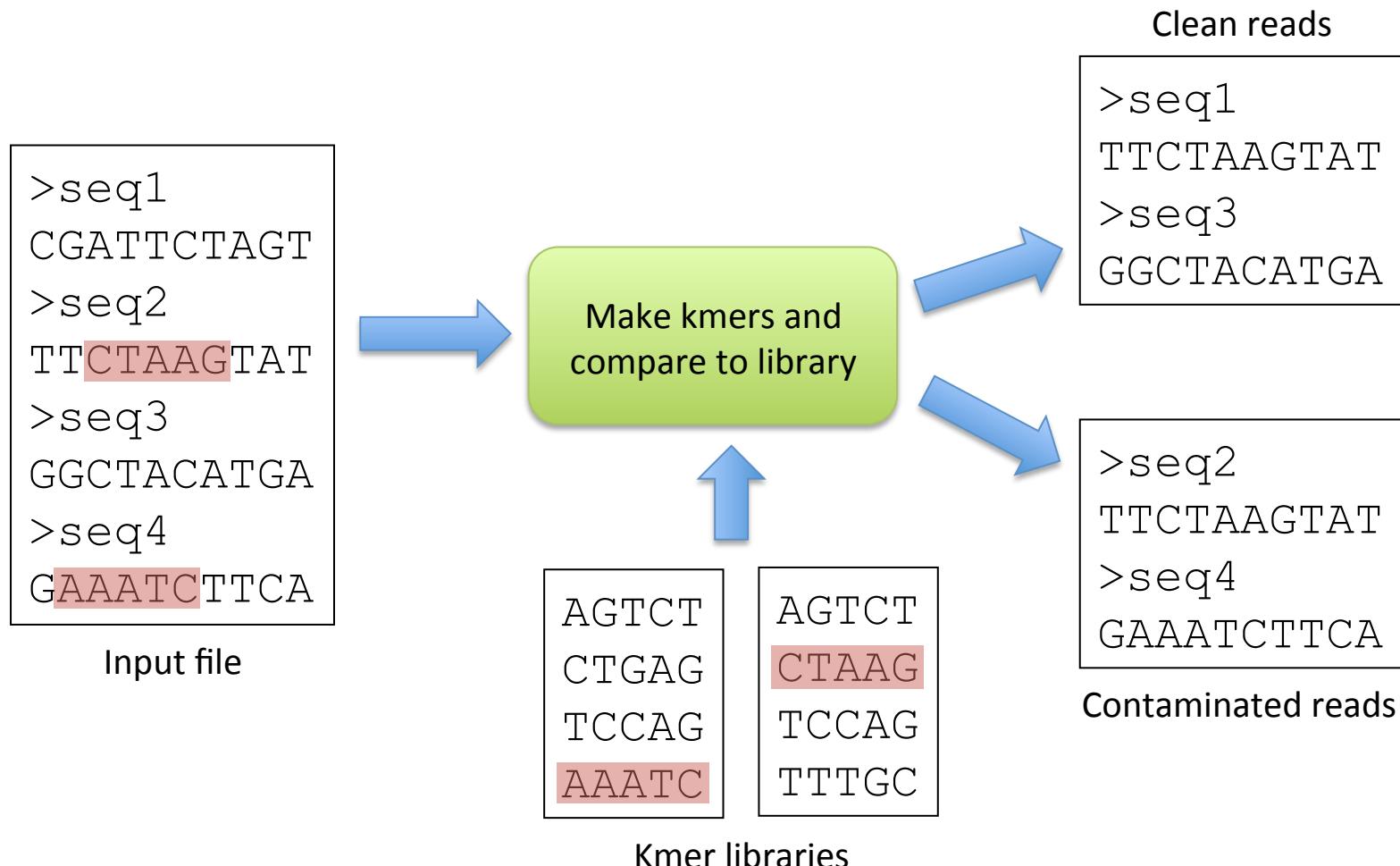
- Remove contaminants.
- Remove host.
- Simplify analysis.

Contamination screening

1. Alignment against potential contaminants or host references
 - e.g. BWA, BLAST
 - Remove reads with good alignment
2. Kontaminant, a TGAC-developed tool for kmer-based screening and filtering...

Kontaminant

- Matches kmers in reads to pre-computed reference:

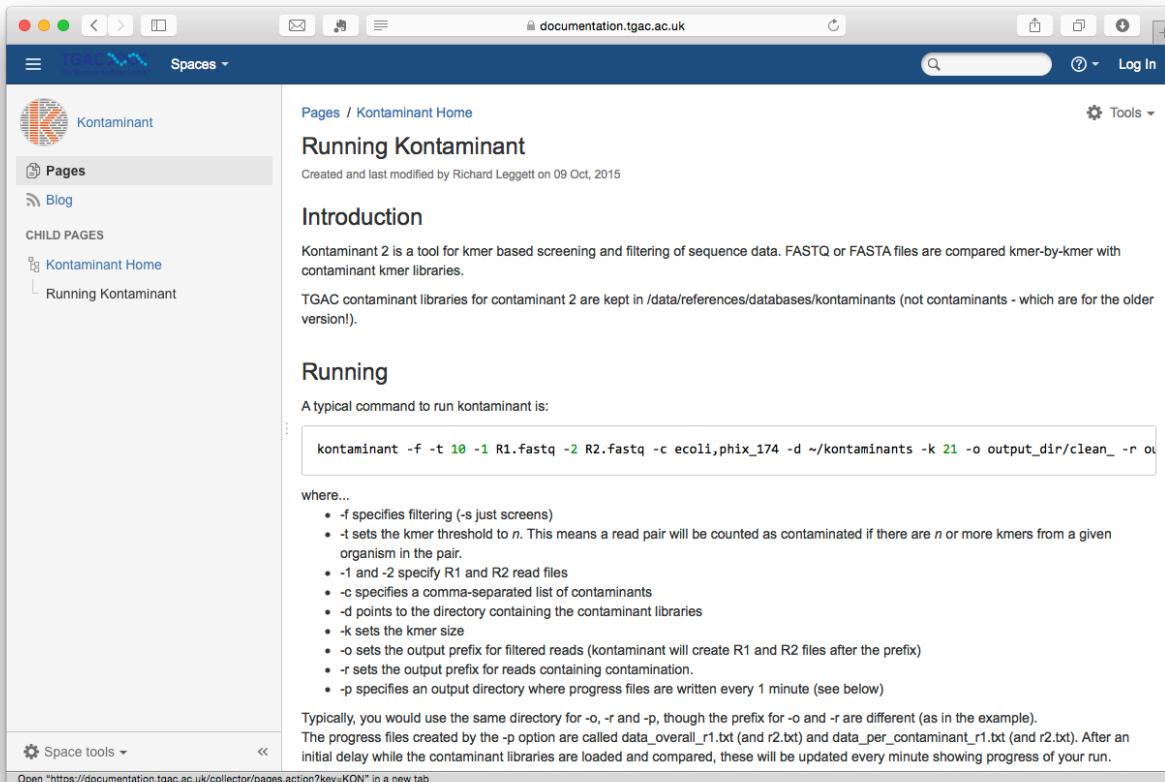


Kontaminant

- Example:

```
kontaminant -f -1 R1.fastq -2 R2.fastq -c ecoli -k 21  
          -o outdir/kept_ -r outdir/removed_
```

- <https://documentation.tgac.ac.uk/>



The screenshot shows a web browser window with the URL <https://documentation.tgac.ac.uk/> in the address bar. The page is titled "Pages / Kontaminant Home". The main content area is titled "Running Kontaminant" and was last modified on 09 Oct, 2015. It includes sections for "Introduction" and "Running". The "Introduction" section explains that Kontaminant 2 is a tool for kmer based screening and filtering of sequence data. The "Running" section provides a typical command to run kontaminant:

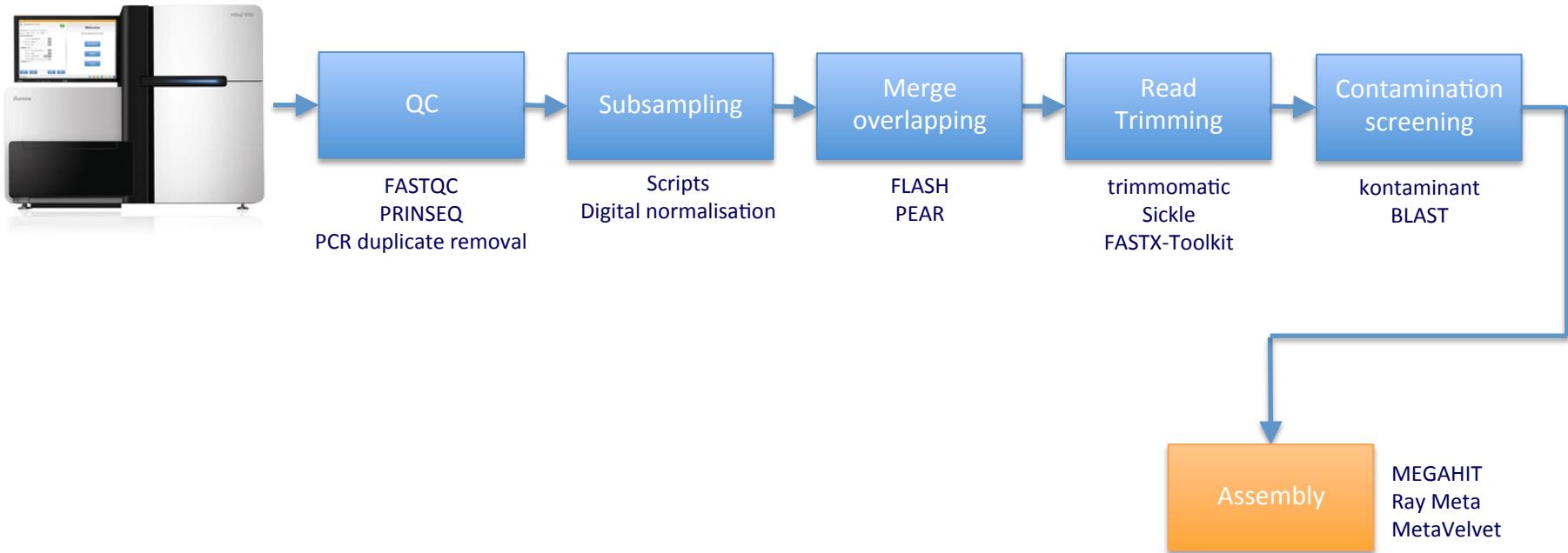
```
kontaminant -f -t 10 -1 R1.fastq -2 R2.fastq -c ecoli,phix_174 -d ~/kontaminants -k 21 -o output_dir/clean_ -r ou
```

Below the command, a list of parameters is provided:

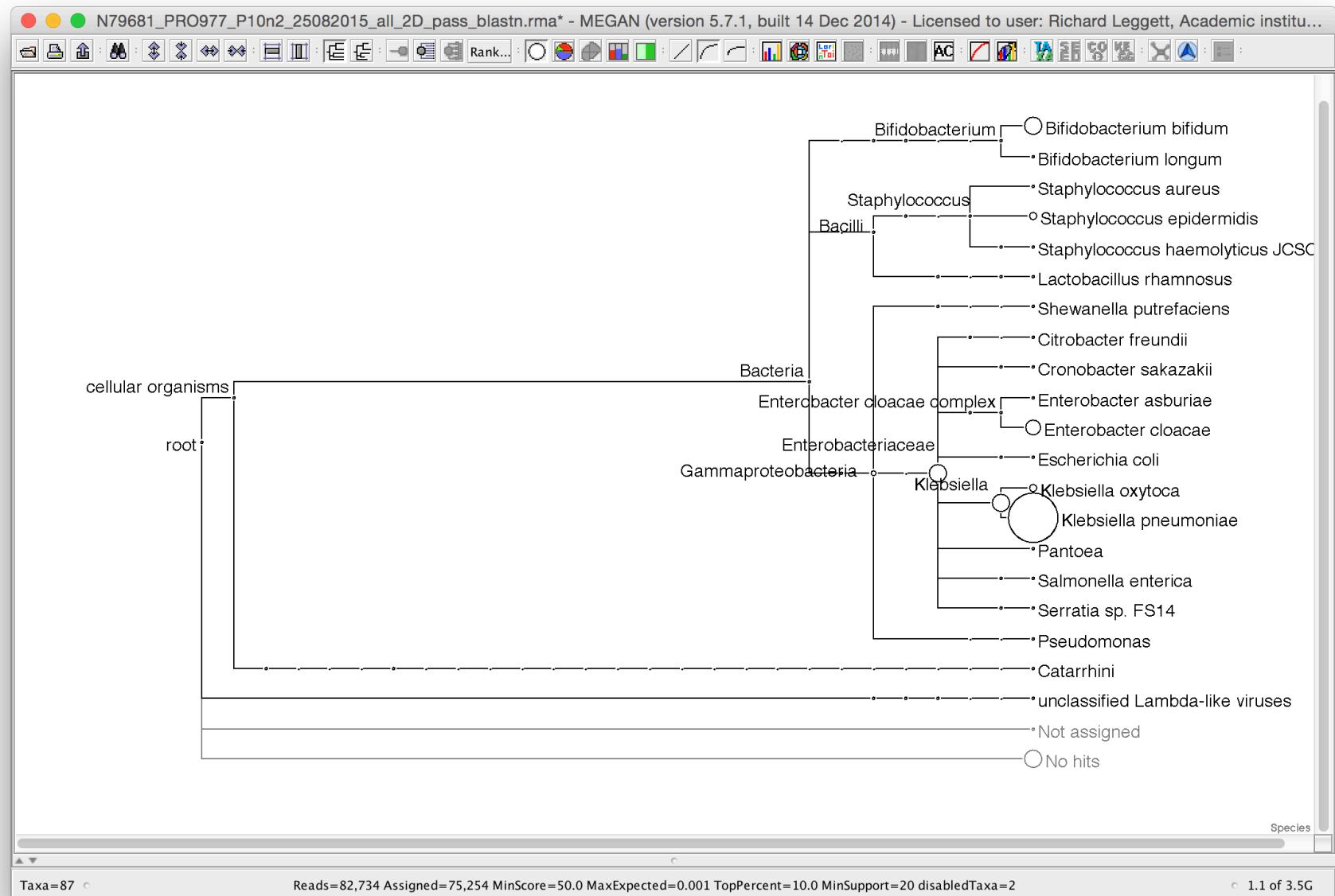
- -f specifies filtering (-s just screens)
- -t sets the kmer threshold to *n*. This means a read pair will be counted as contaminated if there are *n* or more kmers from a given organism in the pair.
- -1 and -2 specify R1 and R2 read files
- -c specifies a comma-separated list of contaminants
- -d points to the directory containing the contaminant libraries
- -k sets the kmer size
- -o sets the output prefix for filtered reads (kontaminant will create R1 and R2 files after the prefix)
- -r sets the output prefix for reads containing contamination.
- -p specifies an output directory where progress files are written every 1 minute (see below)

At the bottom of the page, it notes that typically you would use the same directory for -o, -r and -p, though the prefix for -o and -r are different (as in the example). It also mentions that progress files are created by the -p option and are updated every minute.

Assembly



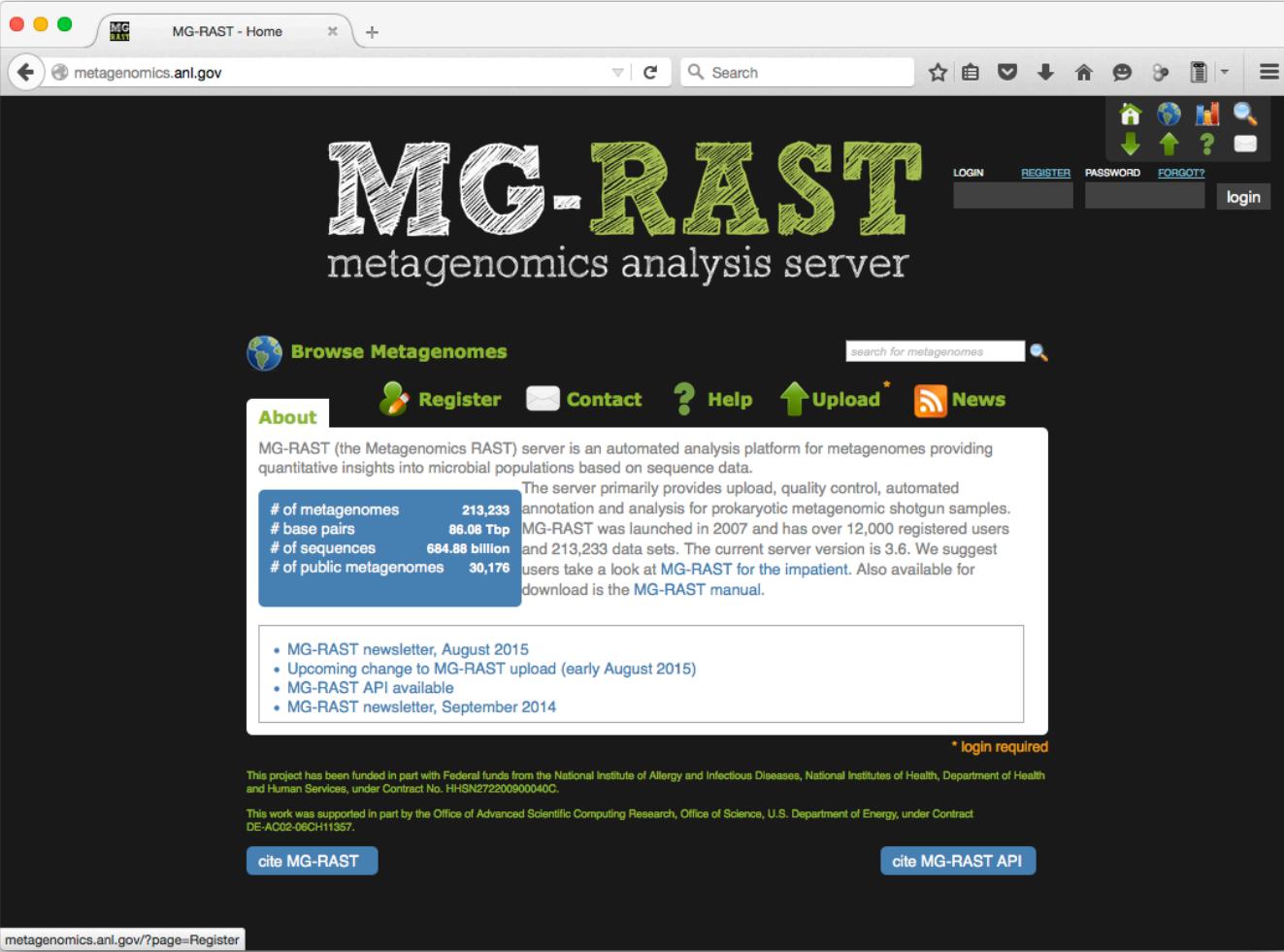
Coming up...



Web portals

- A number of web-based portals exist where you can upload your data and perform analysis.
- Examples:
 - MG-RAST
 - EBI Metagenomics

- <http://metagenomics.anl.gov>



The screenshot shows the MG-RAST Home page. At the top, there's a navigation bar with icons for search, refresh, and other site functions. Below it is a large title "MG-RAST" with "metagenomics analysis server" underneath. To the right of the title are links for "LOGIN", "REGISTER", "PASSWORD", "FORGOT?", and a "login" button. On the left, there's a "Browse Metagenomes" button with a globe icon. In the center, there's a search bar with the placeholder "search for metagenomes" and a magnifying glass icon. Below the search bar are buttons for "About", "Register", "Contact", "Help", "Upload", and "News". A blue box on the left contains statistics: "# of metagenomes 213,233", "# base pairs 86.08 Tbp", "# of sequences 684.88 billion", and "# of public metagenomes 30,176". To the right of these stats is a text block about the server's purpose and history. At the bottom, there's a list of recent news items and a note that login is required. The footer includes funding information from the National Institute of Allergy and Infectious Diseases, National Institutes of Health, Department of Health and Human Services, and the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy.

MG-RAST - Home

metagenomics.anl.gov

Search

MG-RAST

metagenomics analysis server

Browse Metagenomes

search for metagenomes

About Register Contact Help Upload News

of metagenomes 213,233
base pairs 86.08 Tbp
of sequences 684.88 billion
of public metagenomes 30,176

The server primarily provides upload, quality control, automated annotation and analysis for prokaryotic metagenomic shotgun samples. MG-RAST was launched in 2007 and has over 12,000 registered users and 213,233 data sets. The current server version is 3.6. We suggest users take a look at MG-RAST for the impatient. Also available for download is the MG-RAST manual.

- MG-RAST newsletter, August 2015
- Upcoming change to MG-RAST upload (early August 2015)
- MG-RAST API available
- MG-RAST newsletter, September 2014

* login required

This project has been funded in part with Federal funds from the National Institute of Allergy and Infectious Diseases, National Institutes of Health, Department of Health and Human Services, under Contract No. HHSN272200900040C.

This work was supported in part by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract DE-AC02-06CH11357.

cite MG-RAST

cite MG-RAST API

metagenomics.anl.gov/?page=Register

TAXONOMIC HITS DISTRIBUTION [HIDE](#)

The pie charts below illustrate the distribution of taxonomic domains, phyla, and orders for the annotations. Each slice indicates the percentage of reads with predicted proteins and ribosomal RNA genes annotated to the indicated taxonomic level. This information is based on all the annotation source databases used by MG-RAST. An interactive Krona chart of the full taxonomy is also available. Click on a slice or legend to view all sequences annotated with the indicated taxonomic level in the analysis page.

[View taxonomic interactive chart](#)

domain [Download chart data](#)



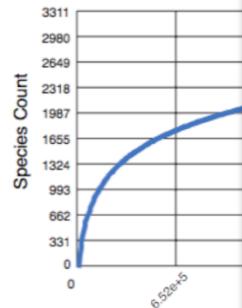
RAREFACTION CURVE [HIDE](#)

The plot below shows the rarefaction curve of annotated species richness. This curve is a plot of the total number of distinct species annotations as a function of the number of sequences sampled. On the left, a steep slope indicates that a large fraction of the species diversity remains to be discovered. If the curve becomes flatter to the right, a reasonable number of individuals is sampled: more intensive sampling is likely to yield only few additional species.

Sampling curves generally rise very steeply at first. Rarefaction curves are calculated from the complete dataset.

[Download chart data](#)

The image is currently dynamic. To



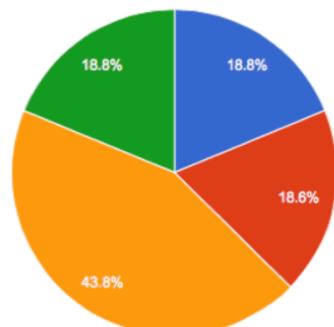
FUNCTIONAL CATEGORY HITS DISTRIBUTION [\[?\]](#) [HIDE](#)

The pie charts below illustrate the distribution of functional categories for at the highest level supported by these functional hierarchies. Each slice indicates the percentage of reads with predicted protein functions annotated to the category for the given source. An interactive Krona chart of each functional hierarchy is also available.

Click on a slice or legend to view all sequences annotated with the indicated category in the analysis page.

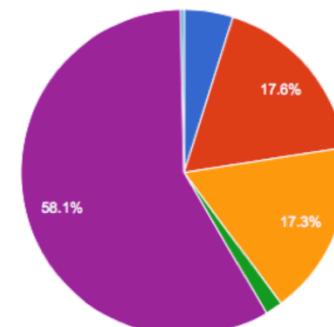
COG [Download chart data](#)

has 74,908 predicted functions
23.2% of predicted proteins
63.7% of annotated proteins
[View COG interactive chart](#)



KO [Download chart data](#)

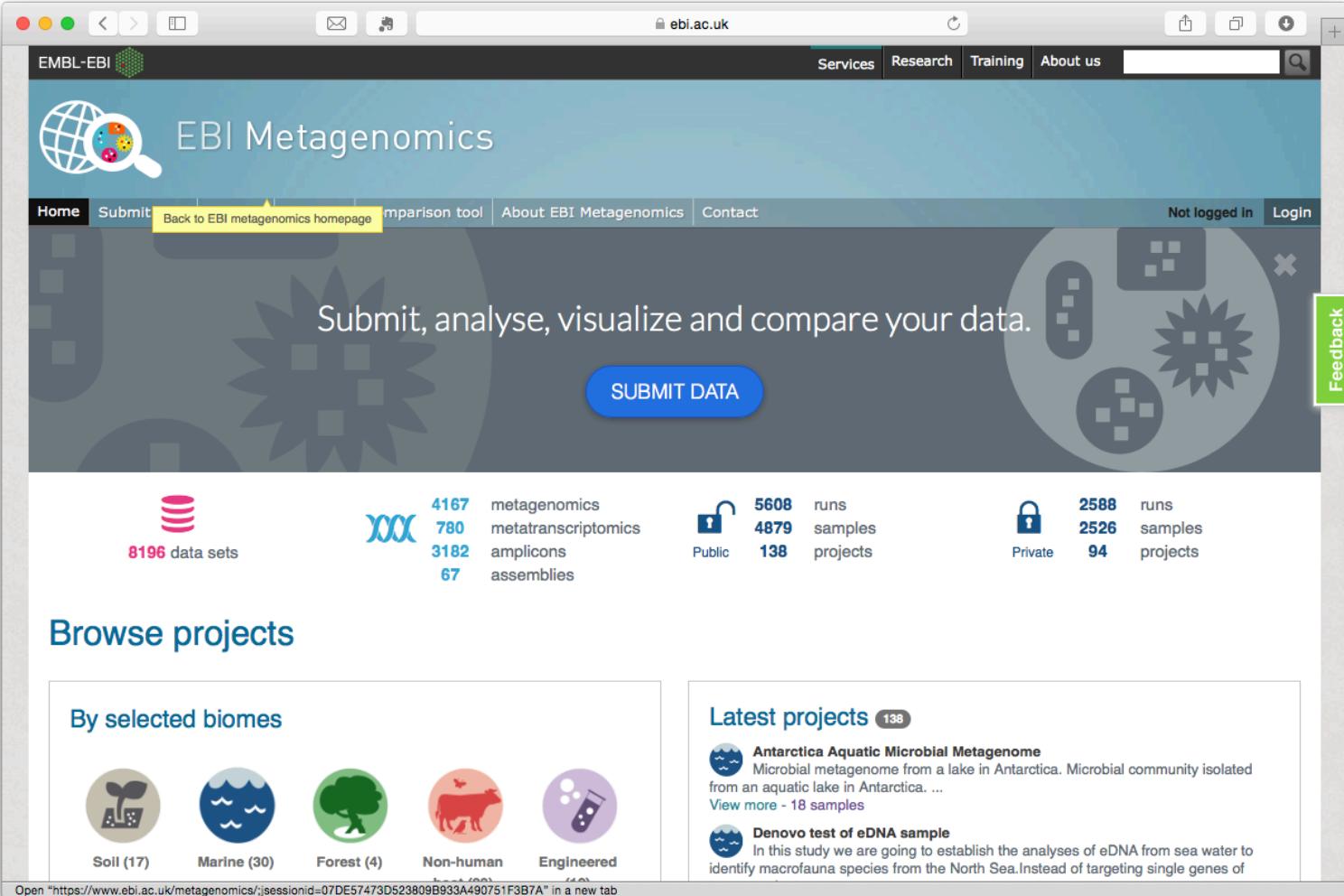
has 44,728 predicted functions
13.9% of predicted proteins
38.0% of annotated proteins
[View KO interactive chart](#)



- Cellular Processes
- Environmental Information Processing
- Genetic Information Processing
- Human Diseases
- Metabolism
- Organismal Systems

EBI Metagenomics

- <https://www.ebi.ac.uk/metagenomics/>



The screenshot shows the EBI Metagenomics homepage. At the top, there's a navigation bar with links for Home, Submit, Back to EBI metagenomics homepage (which is highlighted), Comparison tool, About EBI Metagenomics, Contact, Services, Research, Training, About us, and a search bar. Below the navigation is a large banner with the text "Submit, analyse, visualize and compare your data." and a "SUBMIT DATA" button. To the right of the banner is a "Feedback" link. The main content area features several statistics: 8196 data sets, 4167 metagenomics samples, 780 metatranscriptomics samples, 3182 amplicons, 67 assemblies, 5608 runs, 4879 samples, 138 projects (under Public), 2588 runs, 2526 samples, and 94 projects (under Private). Below this, there's a section titled "Browse projects" with a "By selected biomes" grid showing icons for Soil, Marine, Forest, Non-human, and Engineered environments. To the right, there's a "Latest projects" section with two entries: "Antarctica Aquatic Microbial Metagenome" and "Denovo test of eDNA sample".

EBI Metagenomics

Submit [Back to EBI metagenomics homepage](#) Comparison tool About EBI Metagenomics Contact Services Research Training About us

Not logged in [Login](#)

Home

Submit

8196 data sets

4167 metagenomics samples

780 metatranscriptomics samples

3182 amplicons

67 assemblies

5608 runs

4879 samples

138 projects

2588 runs

2526 samples

94 projects

Public

Private

Browse projects

By selected biomes

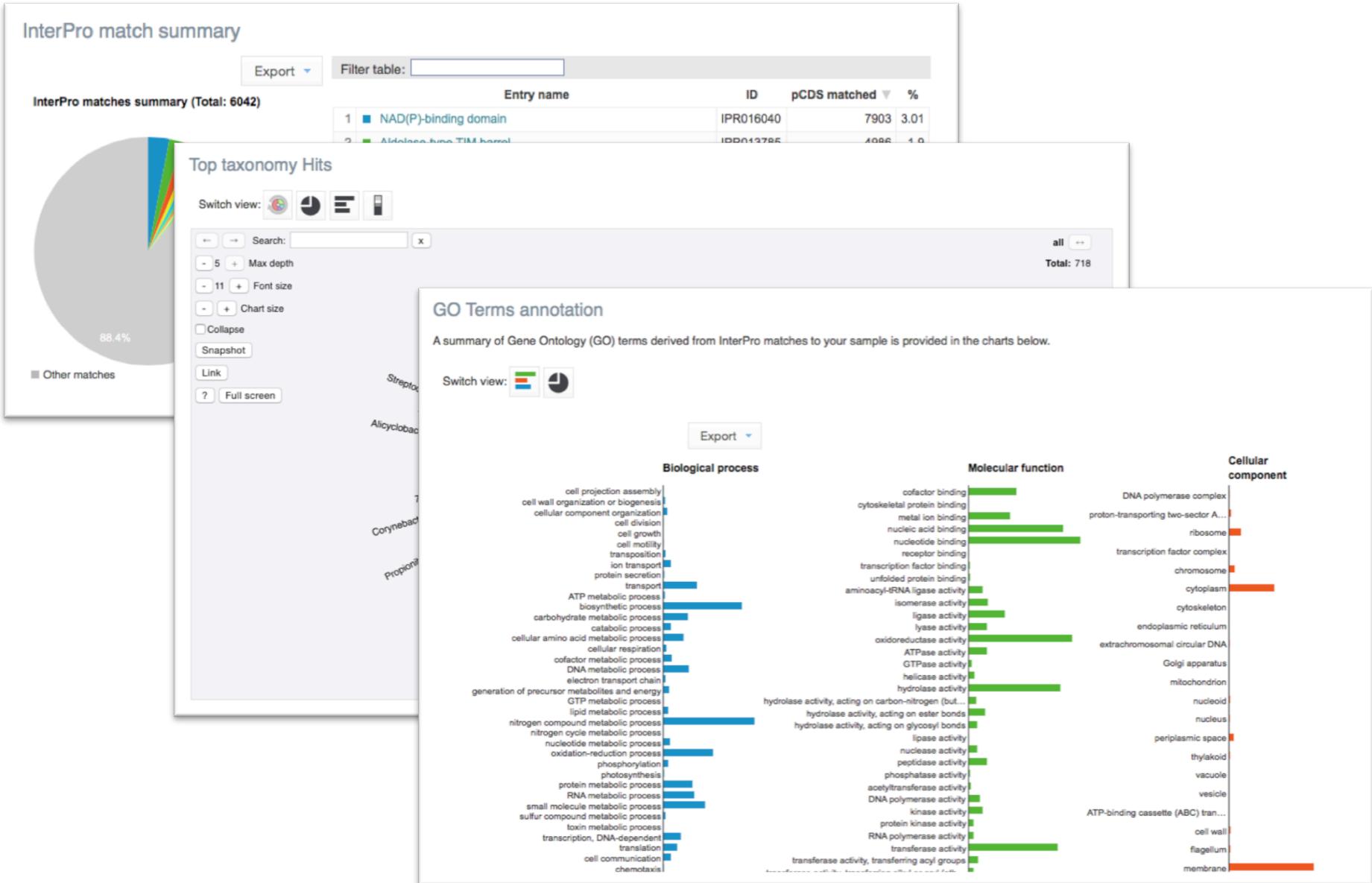
Soil (17) Marine (30) Forest (4) Non-human Engineered

Latest projects 138

 **Antarctica Aquatic Microbial Metagenome**
Microbial metagenome from a lake in Antarctica. Microbial community isolated from an aquatic lake in Antarctica. ...
[View more - 18 samples](#)

 **Denovo test of eDNA sample**
In this study we are going to establish the analyses of eDNA from sea water to identify macrofauna species from the North Sea. Instead of targeting single genes of

InterPro match summary



The End

Thank you for listening.