UNIVERSITEIT STELLENBOSCH UNIVERSITY

Program

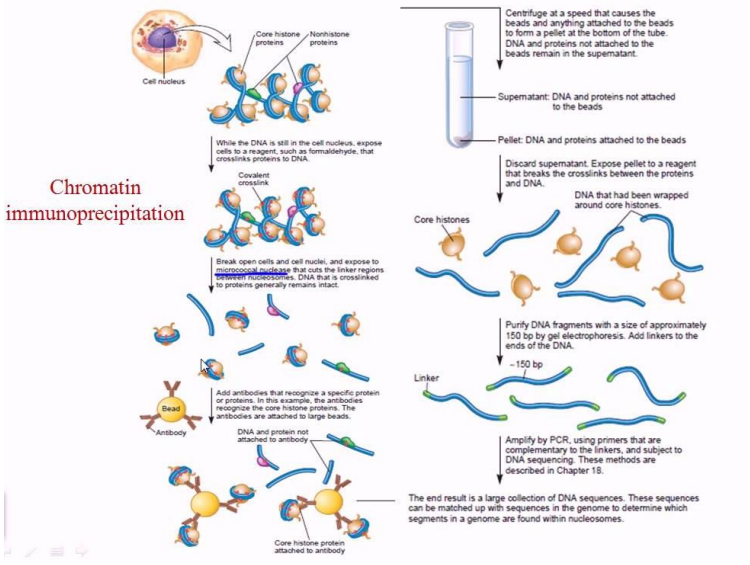
- Welcome
- Introduction: ChIP-seq, NGS sequencing
- FASTQ file format
- Introduction to Galaxy
  - Registering
  - Get Data to Galaxy
  - Histories
  - Tools Menu
- Quality Control: FastQC, Trim_galore
- Mapping: Bowtie2
- Visualizing mapped sequences: IGB
- Peak calling: MACS
- Visualizing peaks
- Computing on genomic intervals: some tools
- Identifying recognition sequences

UNIVERSITEIT STELLENBOSCH UNIVERSITY

- Welcome
- Introduction: ChIP-seq, NGS sequencing
- FASTQ file format
- Introduction to Galaxy
  - Registering
  - Get Data to Galaxy
  - Histories
  - Tools Menu
- Quality Control: FastQC, Trim_galore
- Mapping: Bowtie2
- Visualizing mapped sequences: IGB
- Peak calling: MACS
- Visualizing peaks
- Computing on genomic intervals: some tools
- Identifying recognition sequences

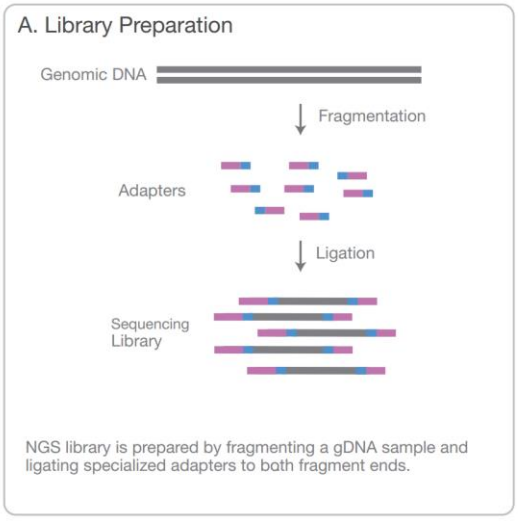
UNIVERSITEIT STELLENBOSCH UNIVERSITY

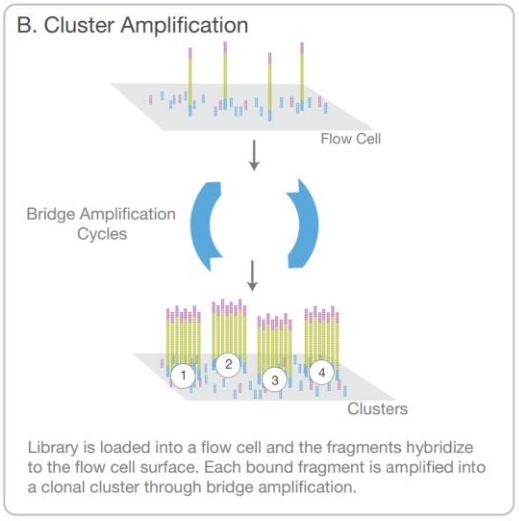
The ChIP-seq technique



UNIVERSITEIT STELLENBOSCH UNIVERSITY

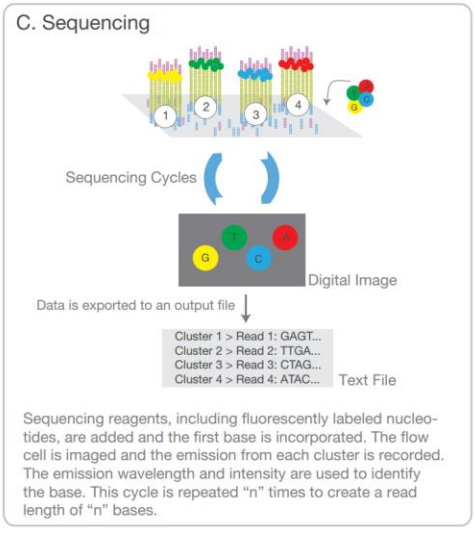
NGS by synthesis (Illumina)





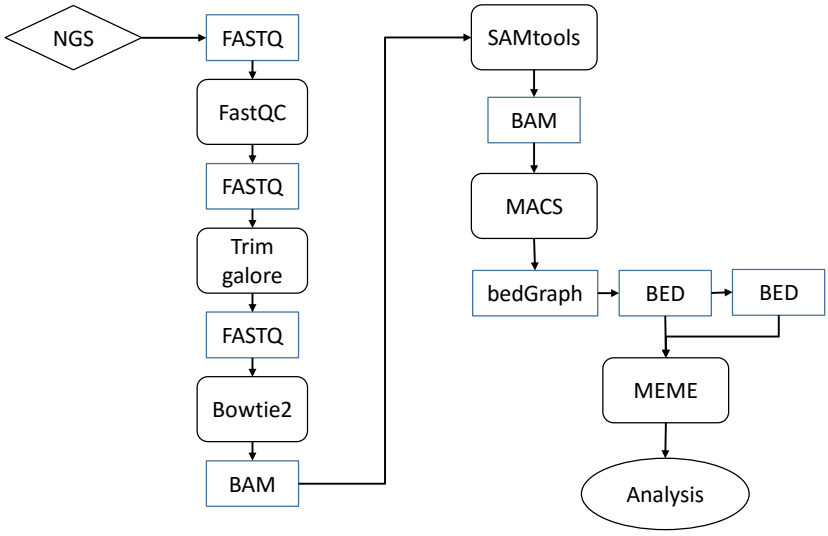
UNIVERSITEIT STELLENBOSCH UNIVERSITY

NGS by synthesis (Illumina)



UNIVERSITEIT STELLENBOSCH UNIVERSITY

ChIP-seq workflow



UNIVERSITEIT STELLENBOSCH UNIVERSITY

- Welcome
- Introduction: ChIP-seq, NGS sequencing
- FASTQ file format
- Introduction to Galaxy
  - Registering
  - Get Data to Galaxy
  - Histories
  - Tools Menu
- Quality Control: FastQC, Trim_galore
- Mapping: Bowtie2
- Visualizing mapped sequences: IGB
- Peak calling: MACS
- Visualizing peaks
- Computing on genomic intervals: some tools
- Identifying recognition sequences

UNIVERSITEIT STELLENBOSCH UNIVERSITY

```
@HWI-D00466:62:C6UETANXX:5:1101:1270:2163 1:N:0:CGATGT
TGTCAGTATAAAAAAATTTTCCGCAGGATATAGAAAAAAAAGAAATGAAATTATAGTAGCGGTTATTTCCGTGGGGTGCTTTTTTACACCTGTACATCTGTTCCCTCCGTACATTTTTTTTATTT
+
BBBBBFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBF//77BBF<FFFFFFFFFFFFFB/<F/BFFFFFFBFFBFFB<BFFB<BB
@HWI-D00466:62:C6UETANXX:5:1101:1332:2180 1:N:0:CGATGT
GCCAATGAAGAAAATACGATGAAACCATGGCAGATCGGAAGAGCACACGTCTGAACTCCAGTCACCGATGTATCTCGTATGCCGTCTTCTGCTTGAAAAAAAAAAATAAACCGACGGTCGCATCA
+
BBBBBFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBBFFFFFFFFFFFFFFFFF<FFF<//FFFFFFFFFFBBBFFFFFFFFFFB<F##########################
@HWI-D00466:62:C6UETANXX:5:1101:4490:2229 1:N:0:CGATGT
GAATCTTATTATTTTCTTTATTTAAATTTATAAAAATATAAAGTCCCCGCCCCCTTTTTATTTTATTTAATTAAGAAGGTATTTTAAAAAAGGAGTGAGGGACCCCCTCCCGTTAGGGAGGGGGA
+
BBBBBFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBF<FFF<FFFFFFFFFFFFFFFFFFFFBFFFFFFFFFFFFFFFFFFFFFFFFFFFBFFFFFFFFFBFFFFFFFFFFFFFF
```

The FASTQ file is a text file composed of blocks of text, with each block containing the following:

- @Sequence name
- Sequence
- +Sequence name (may be absent)
- Phred quality score

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Phred score

The Phred score (Q) is a quantity, where the probability (p) that the assignment is incorrect is given by:

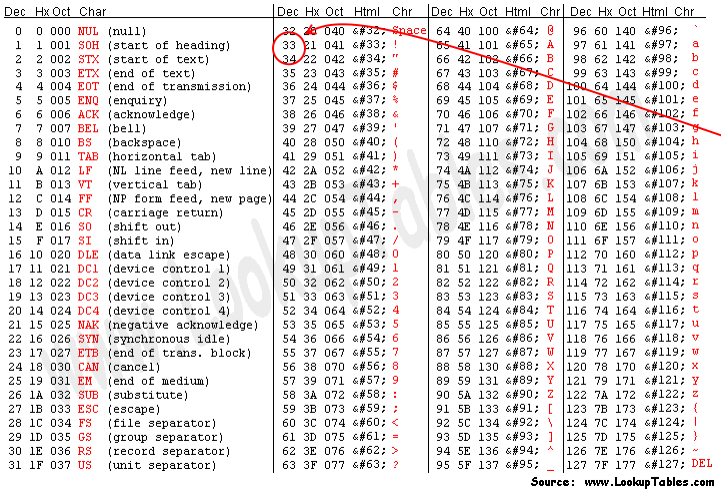
$p = 10^{\frac{-Q}{10}}$

$\therefore$ if $Q = 30, p = 10^{\frac{-30}{10}} = 10^{-3} = 0.001$
$\therefore$ a 1 in 1000 chance that the assigned base is incorrect

Why are Phred scores given as letters?

UNIVERSITEIT STELLENBOSCH UNIVERSITY

The classic 7 bit (0-127) ASCII table



Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.LookupTables.com

The first "real" character (!) has an ASCII value of 33

UNIVERSITEIT STELLENBOSCH UNIVERSITY

ASCII = American Standard Code for Information Interchange

Phred score table

Accuracy (p)	Phred score (Q)	Q+33	ASCII code
1.0	0	33	!
0.1	10	43	+
0.01	20	53	5
0.001	30	63	?
~0.0002	37	70	F
		↓	
$10^{-9.3}$	93	126 (127 is "DEL")	~

```
@HWI-D00466:62:C6UETANXX:5:1101:1270:2163 1:N:0:CGATGT
TGTCAGTATAAAAAAATTTTCCGCAGGATATAGAAAAAAAAGAAATGAAATTATAGTAGCGGTTATTTCCGTGGGGTGCTTTTTTACACCTGTACATCTGTTCCCTCCGTACATTTTTTTTATTT
+
BBBBBFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBF//77BBF<FFFFFFFFFFFFFB/<F/BFFFFFFBFFBFFB<BFFB<BB
```

Each sequenced nucleotide is assigned a quality score
A Phred score higher than 20 (99% accurate) is generally OK

UNIVERSITEIT STELLENBOSCH UNIVERSITY

- Welcome
- Introduction: ChIP-seq, NGS sequencing
- FASTQ file format
- Introduction to Galaxy
  - Registering
  - Get Data to Galaxy
  - Histories
  - Tools Menu
- Quality Control: FastQC, Trim_galore
- Mapping: Bowtie2
- Visualizing mapped sequences: IGB
- Peak calling: MACS
- Visualizing peaks
- Computing on genomic intervals: some tools
- Identifying recognition sequences

UNIVERSITEIT STELLENBOSCH UNIVERSITY

- Welcome
- Introduction: ChIP-seq, NGS sequencing
- FASTQ file format
- Introduction to Galaxy
  - Registering
  - Get Data to Galaxy
  - Histories
  - Tools Menu
- Quality Control: FastQC, Trim_galore
- Mapping: Bowtie2
- Visualizing mapped sequences: IGB
- Peak calling: MACS
- Visualizing peaks
- Computing on genomic intervals: some tools
- Identifying recognition sequences

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Introduction to Galaxy

https://galaxy.sun.ac.za

Note: this instance of galaxy is only accessible to registered users of the SU HPC

Workshop registrants **from Stellenbosch** can login with their **normal IDs** and **passwords**
Non-Stellenbosch attendees can login with **guest1-guest6**, password **GalaxyGuest**

History for this workshop: https://caf-hpc1.sun.ac.za/galaxy/u/hpatterton/h/chip-seq-workshop-22-june-2016

- Upload data files to galaxy
- Select both R1 and R2 files (R1 is the forward primer, and R2 is the reverse primer)
- If the sequences generated several FASTQ files for a sequence run, they wil be named
- …R1_001.fastq, …R2_001.fastq, R1_002.fastq, …R2_002.fastq, …R1_003.fastq, etc.
- Build dataset pair
- Select FastQC (search for it from listbox)
- Select the dataset pair that you have defined
- Execute FastQC with default settings
- FastQC tutorial: https://www.youtube.com/watch?v=bz93ReOv87Y
- Help on each function in FastQC:
- http://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help/3%20Analysis%20Modules/

UNIVERSITEIT STELLENBOSCH UNIVERSITY

H1_CGATGT_L005_R1_001.fastq FastQC Report

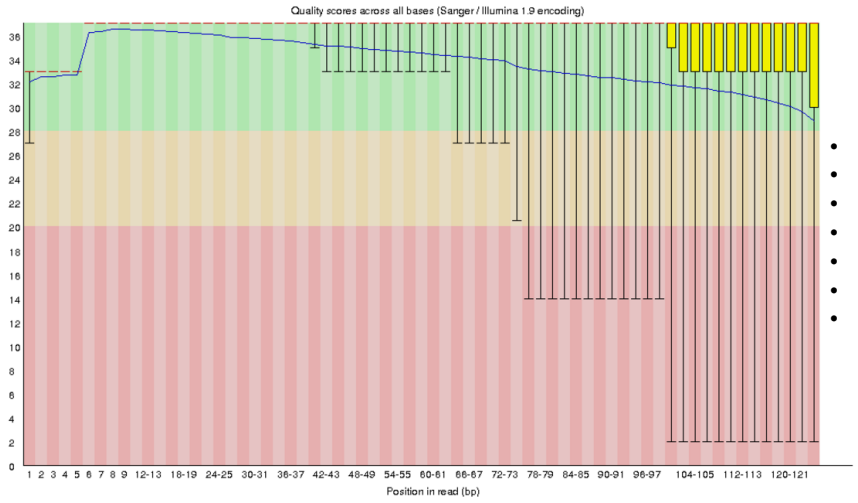
FastQC Report
Thu 9 Jun 2016
H1_CGATGT_L005_R1_001.fastq

Summary

- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content
- Kmer Content

UNIVERSITEIT STELLENBOSCH UNIVERSITY

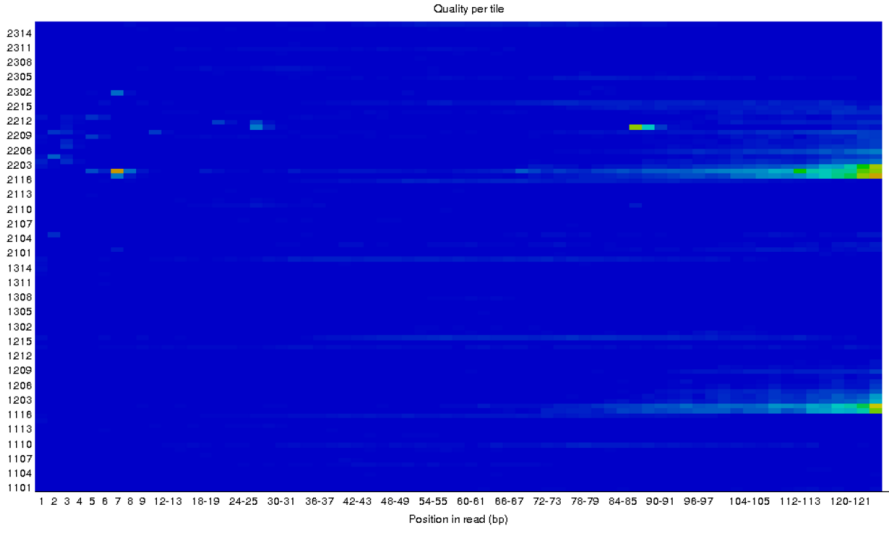
Per base sequence quality



- X-axis: base position
- Y-axis: quality score (Q)
- Yellow box: 25-75th percentile
- Whiskers: 10-90th percentile
- Red line: median
- Blue line: mean
- Ideally, Q > 30

UNIVERSITEIT STELLENBOSCH UNIVERSITY

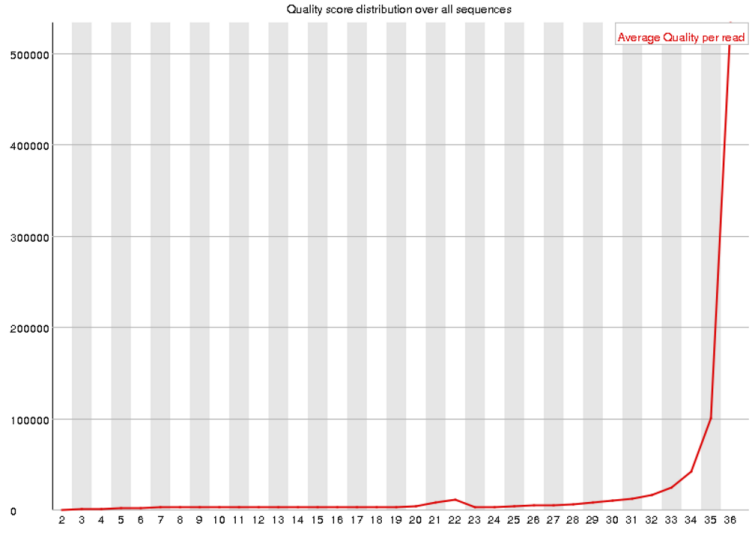
Per tile sequence quality



- Heatmap of read quality at a given position in sequence for each tile in a Illumina flowcell
- Colours displayed from cold (blue) to hot (red)
- Warning issued if any tile has mean Phred score 2 less than flowcell average for that position
- Error issued if any tile has Phred score 5 less than flowcell average at that position

UNIVERSITEIT STELLENBOSCH UNIVERSITY

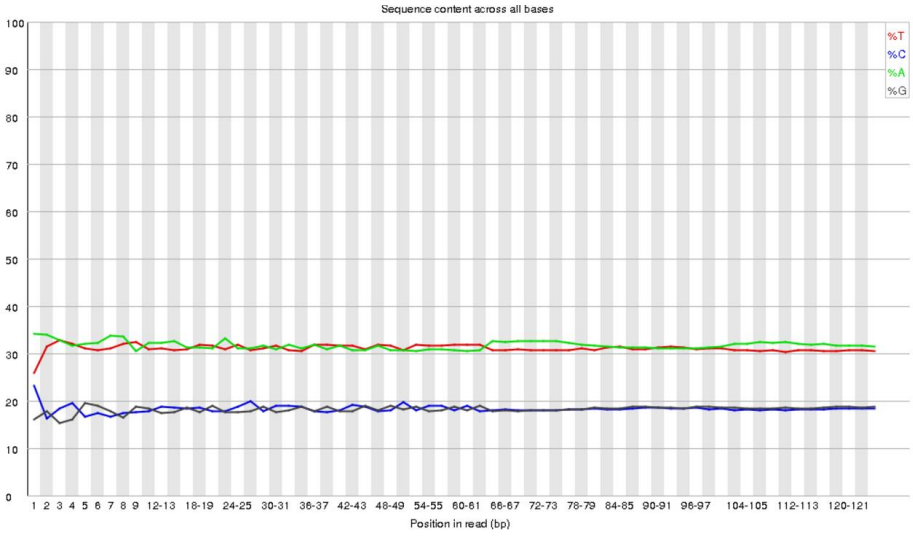
Per sequence quality scores



- Plot of <u>average Q</u> value for a sequence against number of sequences with the same average Q
- Distribution should have pronounced peak at the right, with few if any small bumps in the central/low Q positions

UNIVERSITEIT STELLENBOSCH UNIVERSITY

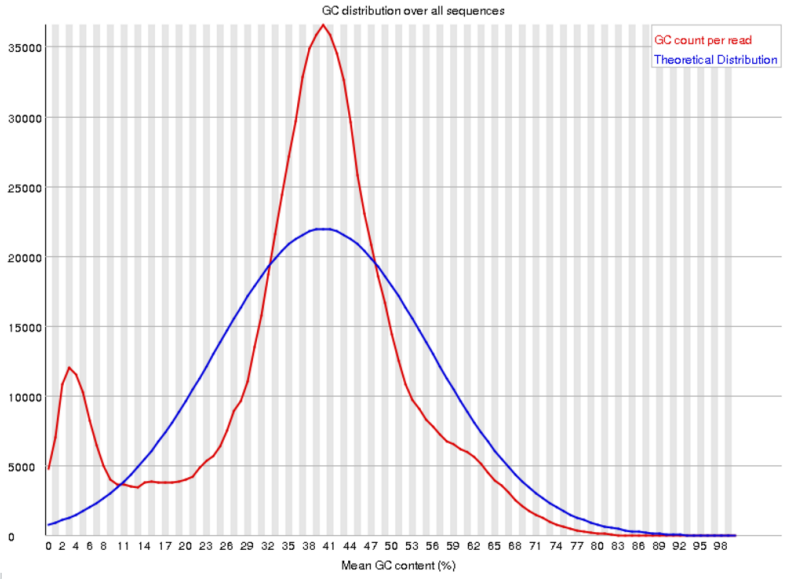
Per base sequence content



- Plot of % content for each of the 4 nucleotides G, A, T and C
- The % composition for each should be fairly constant for the sequence length
- If there is a significant deviation at one or more positions, it may indicate a significant library bias, or problem with the synthesis reaction

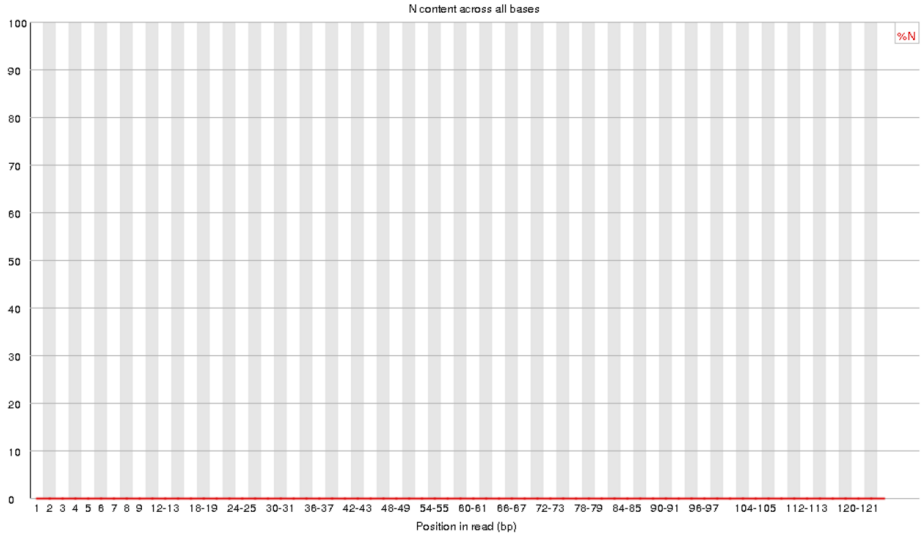
UNIVERSITEIT STELLENBOSCH UNIVERSITY

Per sequence GC content



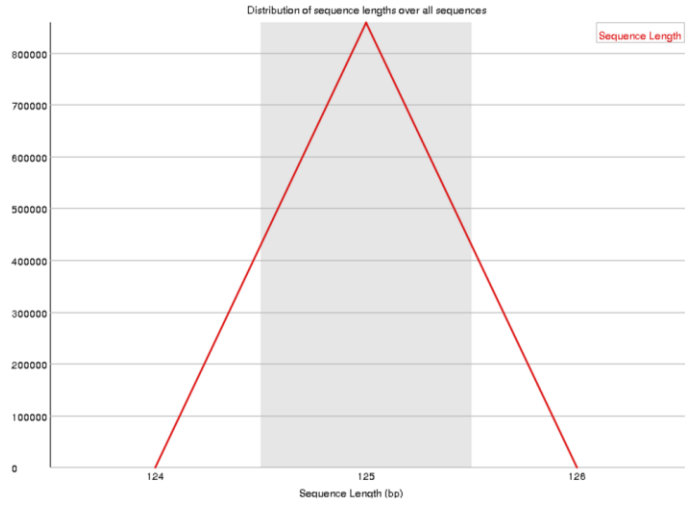
- Plot of the average GC% against number of sequences
- A theoretical, normal distribution is calculated, based on the observed GC% of all the sequences
- The practical distribution and normal distribution should be very similar
- The appearance of "bumps" to the side of the distribution may indicate adapter dimers or another library bias
- A warning is given if the sum of the deviations represent >15% of the reads
- A failure is given if the sum of the deviations represent >30% of the reads

Per base N content



- % of uncalled bases (N) at every position in the sequence
- Indication of sequence quality and base-calling specificity
- Warning issued if N% > 5 at any position
- Failure if N% > 20 at any position

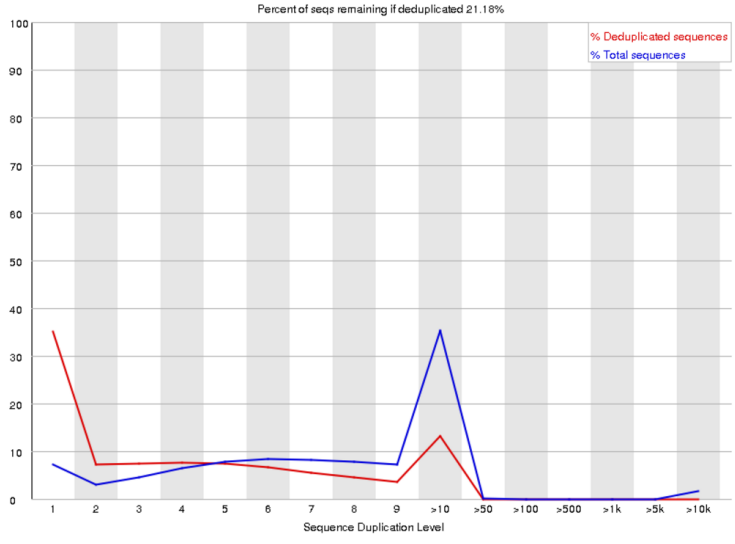
Sequence Length Distribution



- Distribution of sequence lengths in whole dataset
- For Illumina, all sequences should be the same length
- After trimming (see `trim_galore`, later), this distribution may change

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Sequence Duplication Levels



- This plot shown a bin distribution, with an the number of sequences in each bin (blue line)
- Over-representation of sequences may to due to high sequence coverage, or contamination with low-complexity sequences
- Only the first 50 nt of the first 100,000 sequences in a file are analyzed
- The red line shown the “de-duplicated” sequences – each duplicated sequence is counted only once in each bin
- The percentage of sequence remaining after “de-duplication” is given
- Warning: non-unique sequences make up >20% of total
- Error: non-unique sequences make up >50% of total

UNIVERSITEIT STELLENBOSCH UNIVERSITY

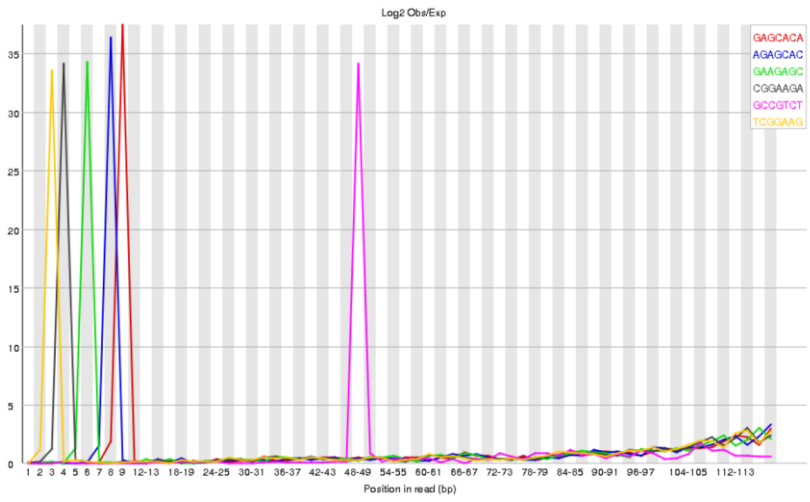
Overrepresented sequences

Sequence	Count	Percentage	Possible Source
GATCGGAAGAGCACACGTCTGAACTCCAGTCACCGATGTATCTCGTATGC	15221	1.7742037006385314	TruSeq Adapter, Index 2 (100% over 50bp)

- The first 100,000 sequences in a file are scanned against the whole file
- A fit over 20 nt with at most one mismatch is a hit
- Hits are screened against a database of common contaminants, including adapter and primer sequences common for the given sequencing platform
- This identified contaminating sequences in the dataset can be removed by filtering the dataset
- Warning: Any one sequence representing >0.1% of dataset
- Error: Any one sequence representing >1% of dataset

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Kmer Content



- The occurrence of each possible k-mer (7-mer) is determined at every position for 2% of the dataset
- The likelihood (p) of finding a specific k-mer at each position is calculated with a binomial distribution
- Top 6 over-represented k-mers shown
- Over-represented sub-sequences are not identified in duplicated sequences or per base content analysis
- Over-represented k-mers may be due to amplification of random primer sub-population

UNIVERSITEIT STELLENBOSCH UNIVERSITY

- Welcome
- Introduction: ChIP-seq, NGS sequencing
- FASTQ file format
- Introduction to Galaxy
  - Registering
  - Get Data to Galaxy
  - Histories
  - Tools Menu
- Quality Control: FastQC, Trim_galore
- Mapping: Bowtie2
- Visualizing mapped sequences: IGB
- Peak calling: MACS
- Visualizing peaks
- Computing on genomic intervals: some tools
- Identifying recognition sequences

UNIVERSITEIT STELLENBOSCH UNIVERSITY

`Trim_galore`

- `Trim_galore` is a wrapper in Galaxy for the `Cutadapt` program
- It is used to trim or remove reads
- Reads trimmed from 3' end where Phred Q is below given value (default=20)
- Dataset is scanned for presence of adapter sequences at 3' end
- The adapter sequences used can be auto-detected or defined as Illumina, Nextera etc.
- Adapter sequences are removed from the 3' end
- User can select how many adapter sequence nucleotides should be present below these are removed (default=1, very stringent)
- User can select to remove N bases from 5' end, or N bases from 3' end
- Remaining sequences less that cutoff length are removed (default<20)
- If dataset is paired-end, both reads will be removed if either is <20 to keep pairs matched in datasets

UNIVERSITEIT STELLENBOSCH UNIVERSITY

`Filter by quality` (Galaxy tool)

- Reads are discarded if the median Q at a percentage of positions is less than a specified Q.

For example:

@CSHL_4_FC042AGOOII:1:2:214:584
GACAATAAAC
+CSHL_4_FC042AGOOII:1:2:214:584
30 30 30 30 30 30 30 30 20 10 *<- converted to integer for clarity*

Using percent = 50 and cut-off = 30 - This read will not be discarded (the median quality is higher than 30).

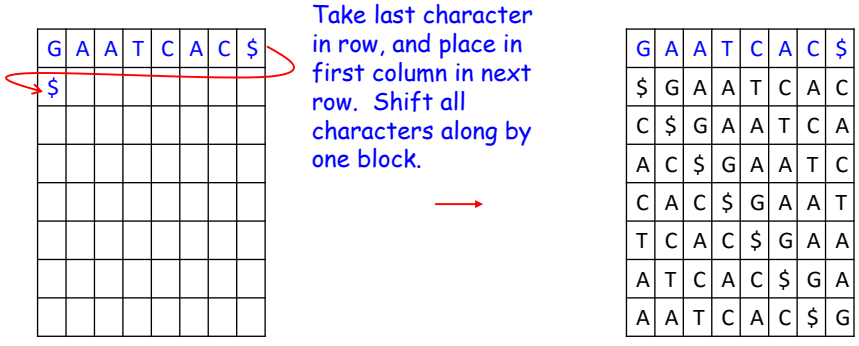
Using percent = 90 and cut-off = 30 - This read will be discarded (90% of the cycles do no have quality equal to / higher than 30).

Using percent = 100 and cut-off = 20 - This read will be discarded (not all cycles have quality equal to / higher than 20).

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Burrows-Wheeler Algorithm (BWA)

- BWA is an algorithm that was originally used in data compression, because it identified repetitive sequence where only the identity of the first character and the number of times it repeats, need to be stored (run length encoding)
- It is based on a "rotating" permutation of a sequence such as GAATCAC ($ is a character that does not occur anywhere else in the sequence):



G	A	A	T	C	A	C	$
$	G	A	A	T	C	A	C
C	$	G	A	A	T	C	A
A	C	$	G	A	A	T	C
C	A	C	$	G	A	A	T
T	C	A	C	$	G	A	A
A	T	C	A	C	$	G	A
A	A	T	C	A	C	$	G

Burrows-Wheeler Algorithm (BWA)

- Sort the rotated matrix lexicographically. The character "$" must have the highest priority

G	A	A	T	C	A	C	$
$	G	A	A	T	C	A	C
C	$	G	A	A	T	C	A
A	C	$	G	A	A	T	C
C	A	C	$	G	A	A	T
T	C	A	C	$	G	A	A
A	T	C	A	C	$	G	A
A	A	T	C	A	C	$	G

Sort on first character

$	G	A	A	T	C	A	C
A	A	T	C	A	C	$	G
A	C	$	G	A	A	T	C
A	T	C	A	C	$	G	A
C	$	G	A	A	T	C	A
C	A	C	$	G	A	A	T
G	A	A	T	C	A	C	$
T	C	A	C	$	G	A	A

Take the last row

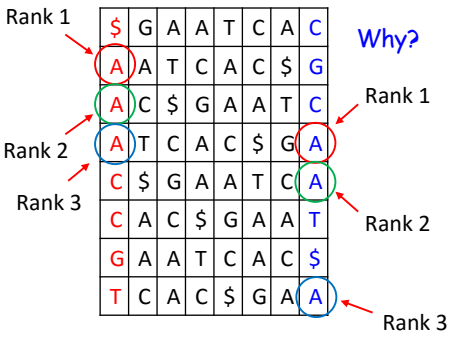
CGCAATA$

Burrows-Wheeler transform BWT(T)

Burrows-Wheeler matrix

Reverse mapping: generating T from BWT(T)

- The original sequence from which the BWT(T) has been derived, can be re-created from the BWT(T).
- This can be demonstrated by giving each character a "rank", i.e., the number of identical characters already encountered in the same column

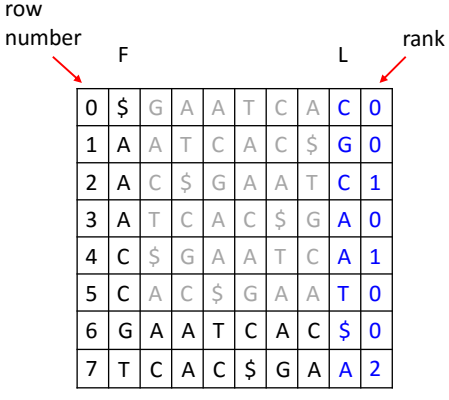


$	G	A	A	T	C	A	C
A	A	T	C	A	C	$	G
A	C	$	G	A	A	T	C
A	T	C	A	C	$	G	A
C	$	G	A	A	T	C	A
C	A	C	$	G	A	A	T
G	A	A	T	C	A	C	$
T	C	A	C	$	G	A	A

- The rows in M are sorted lexicographically
- The 3 A's in column 1 (circled) have identical priority values, and the rows are sorted according to the next character (green)
- In rotated rows where the circled A's are in the last column, the rows are sorted according to the same (green) characters, now in column 1 (orange).
- The rank of each A character is maintained, because they have the *identical ranking priority*, and the characters to the right (green/orange) is *already sorted*

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Last-First (LF) mapping



	F							L	rank
0	$	G	A	A	T	C	A	C	0
1	A	A	T	C	A	C	$	G	0
2	A	C	$	G	A	A	T	C	1
3	A	T	C	A	C	$	G	A	0
4	C	$	G	A	A	T	C	A	1
5	C	A	C	$	G	A	A	T	0
6	G	A	A	T	C	A	C	$	0
7	T	C	A	C	$	G	A	A	2

- The LF Mapping principle: the i^{th} occurrence of a character A in the last column (L) has the same *rank* as the i^{th} occurrence of A in the first column (F)

1. Use first (F) and last (L) column indexing
2. Start with row 0 that will contain $ in column F
3. C has rank 0.
4. Move to 1st C in column F (row 4). It contains A1 in column L
5. Move to A1 in F (row 2). It contains C1 in L.
6. Move to C1 (row 5). L contains T0.
7. Move to T0 (row 7). L contain A2.
8. Move to A2 (row 3). L contains A0.
9. Move to A0 (row 1). L contains G0.
10. Move to G0 (row 6). L contains $ (end)

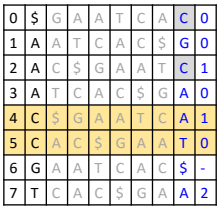
The starting sequence was GAATCAC

Note that LF indexing generates the sequence in reverse

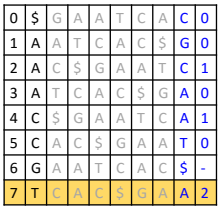
UNIVERSITEIT STELLENBOSCH UNIVERSITY

Finding patterns using a BWT and LF mapping

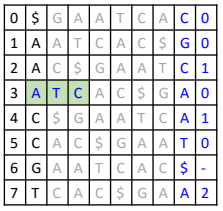
- We want to find the pattern ATC in a sequence



Start with the last character, C
There are two occurrences of C
C is preceded by A1 or T0



There is only 1 T entry
It is preceded by A2

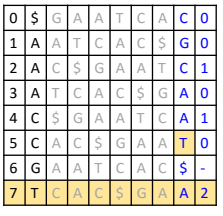


We have found the pattern

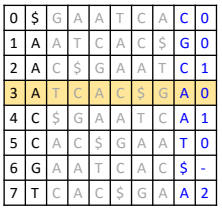
What if the pattern does not occur in the sequence?

Absent patterns

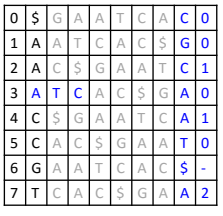
- We want to find the pattern TAT in a sequence



T0 maps to A2



A2 maps to A0



There is no occurrence where an AT is preceded by a T

- If a pattern occurs more than once, we will have more than one possible row when we have moved to the 1st character of the pattern

Finding a matched pattern position in a sequence

- The "brute force" solution after finding the pattern is to continue with the sequence until you find the $ -- this is very inefficient though

0	$	G	A	A	T	C	A	C	0
1	A	A	T	C	A	C	$	G	0
2	A	C	$	G	A	A	T	C	1
3	A	T	C	A	C	$	G	A	0
4	C	$	G	A	A	T	C	A	1
5	C	A	C	$	G	A	A	T	0
6	G	A	A	T	C	A	C	$	-
7	T	C	A	C	$	G	A	A	2

A2→A0
A0 →G0
G0 →$

∴ 2 steps → offset 2 from start

GAATCAC

But what if it is a 1,000,000 bp sequence?

Determining pattern position with a index

original row position in unsorted matrix
also known as the "suffix array" (SA)

$	G	A	A	T	C	A	C	7
A	A	T	C	A	C	$	G	1
A	C	$	G	A	A	T	C	5
A	T	C	A	C	$	G	A	2
C	$	G	A	A	T	C	A	6
C	A	C	$	G	A	A	T	4
G	A	A	T	C	A	C	$	0
T	C	A	C	$	G	A	A	3

- We can see that the row starting with the ATC pattern was 2 rotations (SA=2) from the original sequence start
- ∴ offset 2 from the sequence start
- However, we now have to store an integer list as long as the sequence itself → not very memory efficient!
- What if we throw away part of the suffix array?

Finding an offset with a partial row index list

0	$	G	A	A	T	C	A	C	0	7
1	A	A	T	C	A	C	$	G	0	1
2	A	C	$	G	A	A	T	C	1	
3	A	T	C	A	C	$	G	A	0	
4	C	$	G	A	A	T	C	A	1	6
5	C	A	C	$	G	A	A	T	0	
6	G	A	A	T	C	A	C	$	-	
7	T	C	A	C	$	G	A	A	2	3

- The row in which we have identified the pattern has no SA entry
- However, we can follow the rank A0 to row 1
- Row 1 has a SA entry of 1
- Now, 1 step + SA of 1 = 2 steps to start of sequence

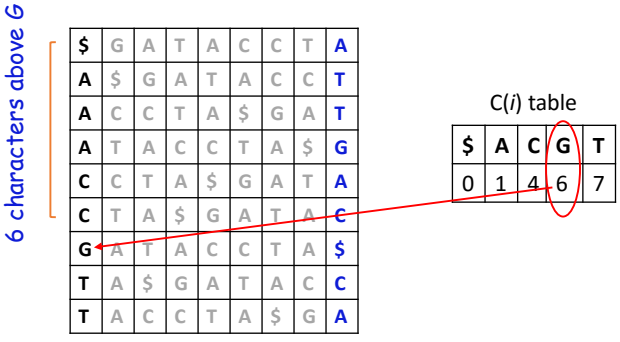
FM-index

- The FM-index (proposed by Ferragina and Manzini) introduces another indexing scheme with performance improvements over the suffix array index
- It is based on the LF(i) mapping principle
- It uses 2 tables:
  - C table
  - Occ table
- The **C** (character) table is really a table with the total number of characters with a lexicographic priority larger than a given character, i.e., the number of characters in the F (sorted) column above the first occurrence of a given character
- The **Occ** table is a table with the rank of each character in each row of the BWT(T)

UNIVERSITEIT STELLENBOSCH UNIVERSITY

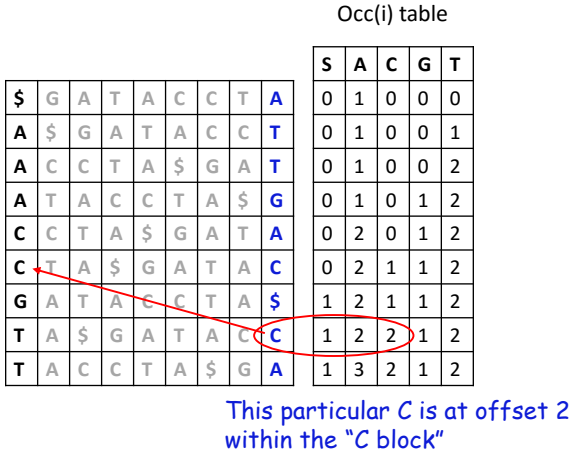
The C(i) table

- The C(i) table gives the *index* of the first A, C, G etc. character in row F, i.e., the *offset* to the character "block"

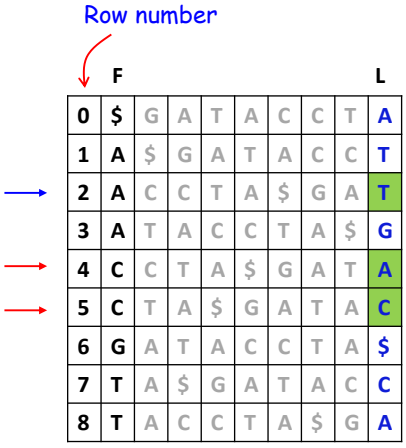


The Occ(i) table

- The Occ(i) table gives the rank of a character, i.e., the *offset* to that character in its "block" in line F



Use of the FM index to find patterns



- To find the sequence TAC, we simply calculate two pointers, start and end, from C(i):
- Start = C("C") = 4; End = C("C+1") = C("G") = 6
- C block ends 1 step before G block starts, ∴ 6-1 = 5
- That points to "A" and "C" in L (red arrows)
- Only A matches the character in the search sequence
- This particular A is at:
- C("A")+Occ("A",row) = 1+2 = 3
- The first character in Occ is index 1 not 0, so the row number is one less: 3-1 = 2 (blue arrow)
- The preceding character is a "T"
- We have found the pattern *without needing the F column*

- ***I.e. we only need the C(i) and Occ(i) tables and the BWT(T) column to search for patterns***

- ***We can similarly discard part of the Occ table, and continue mapping until we do hit a checkpoint to determine the position of the pattern in our large sequence***

UNIVERSITEIT STELLENBOSCH UNIVERSITY

The FM Index allows a small memory footprint

Components of the FM Index:

First column (F): ~ | ∑ | integers
Last column (L): m characters
SA sample: m · a integers, where a is fraction of rows kept
Checkpoints: m × | ∑ | · b integers, where b is fraction of rows check-pointed

Example: DNA alphabet (2 bits per nucleotide), T = *human genome*, a = 1/32, b = 1/128

First column (F):	16 bytes
Last column (L):	2 bits * 3 billion chars = 750 MB
SA sample:	3 billion chars * 4 bytes/char / 32 = ~ 400 MB
Checkpoints:	3 billion * 4 bytes/char / 128 = ~ 100 MB

Total < 1.5 GB

UNIVERSITEIT STELLENBOSCH UNIVERSITY

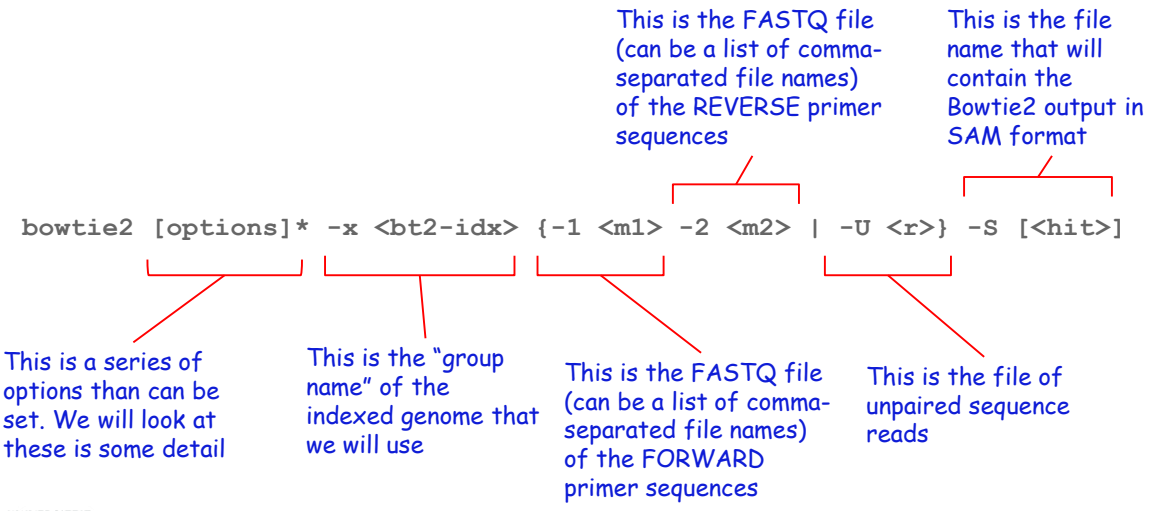
Bowtie 2

- Bowtie 2 is an ultrafast and memory-efficient tool for aligning sequencing reads to long reference sequences.
- It is particularly good at aligning reads of about 50 up to 100s or 1,000s of characters to relatively long (e.g. mammalian) genomes.
- Bowtie 2 indexes the genome with an FM Index (based on the Burrows-Wheeler Transform or BWT) to keep its memory footprint small: for the human genome, its memory footprint is typically around 3.2 gigabytes of RAM.
- Bowtie 2 supports gapped, local, and paired-end alignment modes. Multiple processors can be used simultaneously to achieve greater alignment speed.
- Bowtie 2 outputs alignments in SAM format, enabling interoperation with a large number of other tools (e.g. SAMtools, GATK) that use SAM.

From the Bowtie2 manual

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Briefly looking at command line Bowtie2



UNIVERSITEIT STELLENBOSCH UNIVERSITY

Bowtie2 modes

- The default alignment mode of Bowtie2 is end-to-end. Local alignment is also possible
- The performance/sensitivity setting can also be chosen:

--very-fast	-D 5 -R 1 -N 0 -L 22 -i S,0,2.50
--fast	-D 10 -R 2 -N 0 -L 22 -i S,0,2.50
--sensitive	-D 15 -R 2 -L 22 -i S,1,1.15 (default in --end-to-end mode)
--very-sensitive	-D 20 -R 3 -N 0 -L 20 -i S,1,0.50

-D	number of consecutive seed extension attempts that can "fail"
-R	maximum number of times Bowtie 2 will "re-seed" reads
-N	number of mismatches allowed in a seed alignment
-L	length of the seed substrings to align
-i	function governing the interval between seed substrings -i S,1,2.5 sets the interval function f to f(x) = 1 + 2.5 * sqrt(x) where x is the read length

- The default –sensitive mode is generally suitable for most alignments. Only change if you have a good reason to do so

Bowtie2 options

--n-ceil <func>
Sets a function governing the maximum number of ambiguous characters (usually Ns and/or .s) allowed in a read as a function of read length. For instance, specifying -L,0,0.15 sets the N-ceiling function f to f(x) = 0 + 0.15 * x, where x is the read length.

-I
The minimum fragment length for valid paired-end alignments.

-X
The maximum fragment length for valid paired-end alignments.

--fr/--rf/--ff
The upstream/downstream mate orientations for a valid paired-end alignment against the forward/reference strand.

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Bowtie2 options

- Specify how Bowtie2 should treat alignment pairs that over-lap in different ways

--dovetail
If the mates "dovetail", that is if one mate alignment extends past the beginning of the other such that the wrong mate begins upstream, consider that to be concordant.

--no-contain
If one mate alignment contains the other, consider that to be non-concordant.

--no-overlap
If one mate alignment overlaps the other at all, consider that to be non-concordant.

Alignment reporting

- Bowtie2 will report the best alignment of all alignments found that passes the selection criteria
- If there are several equal “best” alignments, Bowtie will randomly report one

Options

- -k <int> Bowtie will report up to <int> number of alignments for each query sequence, of which some will be poorer, “secondary” alignments
- -a like -k, but there is no upper limit on the number or alignments searched for

Bowtie2 outputs

```
10000 reads; of these:
  10000 (100.00%) were paired; of these:
    650 (6.50%) aligned concordantly 0 times
    8823 (88.23%) aligned concordantly exactly 1 time
    527 (5.27%) aligned concordantly >1 times
    ----
    650 pairs aligned concordantly 0 times; of these:
      34 (5.23%) aligned discordantly 1 time
    ----
    616 pairs aligned 0 times concordantly or discordantly; of these:
      1232 mates make up the pairs; of these:
        660 (53.57%) aligned 0 times
        571 (46.35%) aligned exactly 1 time
        1 (0.08%) aligned >1 times
96.70% overall alignment rate
```


SAM file format

A SAM format file is a text file composed of header lines, and sequence lines
The header lines (if present) are preceded by '@' and a two character code such as @HD (header line), @SQ (Reference sequence dictionary) etc.

The rest of the file is composed of "sequence lines", one after the other. Each sequence line is composed of 11 tab delimited fields:

1 QNAME (Query template NAME)
2 FLAG (bitwise FLAG)
3 RNAME (Reference sequence NAME)
4 POS (1-based leftmost mapping POSition)
5 MAPQ (MAPping Quality)
6 CIGAR (CIGAR string)
7 RNEXT (Ref. name of the mate/next read)
8 PNEXT (Position of the mate/next read)
9 TLEN (observed Template LENgth)
10 SEQ (segment SEQuence)
11 QUAL (Phred-scaled base QUALity+33)

See the SAM file format specification

Sequence string example

1	QNAME	HWI-ST193:439:D16G8ACXX:4:1101:2568:2202
2	FLAG	99
3	RNAME	Tb427_11_01_v4
4	POS	462830
5	MAPQ	49
6	CIGAR	50M
7	RNEXT	=
8	PNEXT	462931
9	TLEN	151
10	SEQ	TGGCTGCATCCTCTTTGCCCAGTTCCTTTGCATCATCATCAGCTGTGGGA
11	QUAL	@@<DFFBBHHHBF@FDGGGGGGEHHIDG@EGAEAGGIIIIFHHHHGG3DH
12	YT:Z:UP	
13	YF:Z:NS	

Q = -10 log10 p, where p is an estimate of the probability that the alignment does **not** correspond to the read's true point of origin

i.e. 49 → $10^{\frac{-49}{10}} = 10^{-4.9} \sim 0.00001$

All optional fields follow the TAG:TYPE:VALUE

UNIVERSITEIT STELLENBOSCH UNIVERSITY

SAM bitwise flag

Bit	0	1	2	3	4	5	6	7	8	9	10
Value	1	2	4	8	16	32	64	128	256	512	1024

Selecting 767 = 256 + 512, and therefore selects bits 8 and 9

1	The read is one of a pair
2	The alignment is one end of a proper paired-end alignment
4	The read has no reported alignments
8	The read is one of a pair and has no reported alignments
16	The alignment is to the reverse reference strand
32	The other mate in the paired-end alignment is aligned to the reverse reference strand
64	The read is mate 1 in a pair
128	The read is mate 2 in a pair
256	Secondary alignment
512	Not passing filters, such as platform/vendor quality controls
1024	PCR or optical duplicate 2048 0x800 supplementary alignment

You will only get secondary alignments if you run Bowtie2 with the -k option

You can use these flags to remove poorly mapped reads from your dataset

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Filter mapped reads by MAPQ

- Samtools is a suite of programs for interacting with high-throughput sequencing data
- SAMtools is composed of many, many individual tools (see the manual)
- SAMtools can be run as a command line tool (on Linux) or is available as individual tools in Galaxy
- A useful tool to filter mapped entries in a SAM/BAM file in terms of a cut-off MAPQ score is:

```
samtools view [options] in.sam|in.bam|in.cram [region...]
```

with option -q <int>	Skip alignments with MAPQ < int
with option –f <value>	only write sequence reads with the bits set in value (must be hex format 0x…) that is also set in the FLAG value of the read example –f

See SAMtools:Filter SAM or BAM

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Samtools:Filestat to see details of mapped reads

```
3679873 + 0 in total (QC-passed reads + QC-failed reads)
0 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
3679873 + 0 mapped (100.00%:-nan%)
3679873 + 0 paired in sequencing
1848347 + 0 read1
1831526 + 0 read2
3643874 + 0 properly paired (99.02%:-nan%)
3656927 + 0 with itself and mate mapped
22946 + 0 singletons (0.62%:-nan%)
405 + 0 with mate mapped to a different chr
405 + 0 with mate mapped to a different chr (mapQ>=5)
```

See Samtools:Filestat

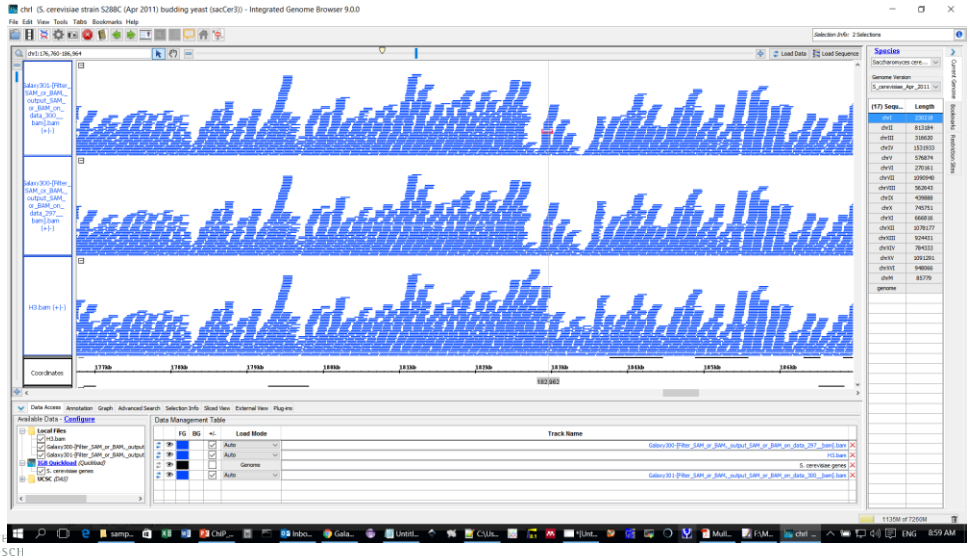
UNIVERSITEIT STELLENBOSCH UNIVERSITY

BAM format files

- A BAM format file is a compressed SAM file
- BAM and SAM format files can be interconverted with SAMtools
- BGZF, a variant of gzip, is used to generate a BAM file
- In a BAM file blocks are stored which can be quickly with an index
- The BAM index file is a BAI file
- Tools that can visualize BAM files (such as Genome Browsers) require the BAI file
- A BAI file can be regenerated from a BAM file

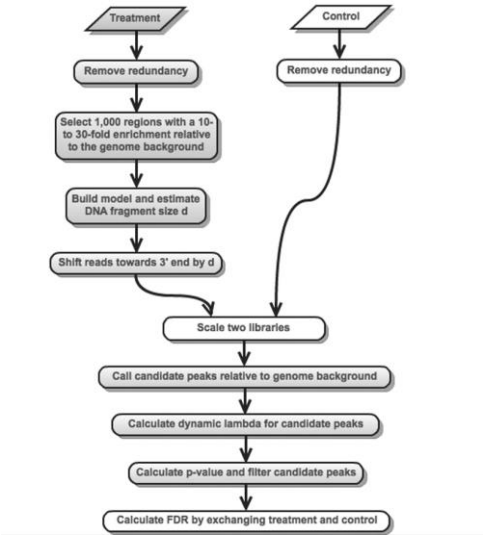
UNIVERSITEIT STELLENBOSCH UNIVERSITY

Visualize mapped reads in IGB



UNIVERSITEIT STELLENBOSCH UNIVERSITY

Finding peaks in mapped datasets: MACS



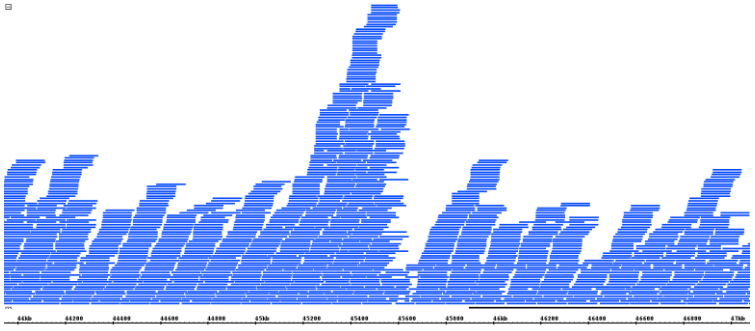
UNIVERSITEIT STELLENBOSCH UNIVERSITY

MACS: 4 steps

- removing redundant reads
- adjusting read position
- calculating peak enrichment
- estimating the empirical false discovery rate (FDR)

UNIVERSITEIT STELLENBOSCH UNIVERSITY

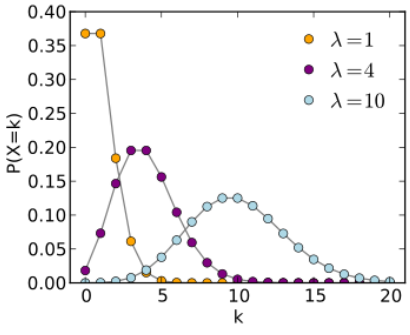
MACS: How to find peaks of enrichment



Is this a peak?

Use a Poisson distribution to define peaks

The Poisson distribution



Poisson distribution:

$$p(X \leq k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

p: probability mass function
λ: average rate of an occurrence in a given interval
k: the number of occurrences tested for

Ex: 2 goals on average per match.
Chance of 4 goals in a match?

A Poisson distribution is a discrete probability distribution that expresses the probability of a given number of events occurring in a fixed interval of time and/or space if these events occur with a known average rate and independently of the time since the last event

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Poisson distribution is linearly proportional over an interval

This means that if the possibility of something occurring over length x = p, the chance of it occurring over 2x = 2p
The possibility of two different observations in the same interval is additive, i.e.:
At a given λ, the chance of making one observation (k=1) or two observations (k=2) is

$$\frac{\lambda^1 e^{-\lambda}}{1} + \frac{\lambda^2 e^{-\lambda}}{2}$$

Or, in the general case:

$$\sum_{k=1}^{N} \frac{\lambda^k e^{-\lambda}}{k!}$$

The chance of making more than N observations would be:

$$1 - \left[\frac{\lambda^0 e^{-\lambda}}{1} + \frac{\lambda^1 e^{-\lambda}}{1} + \frac{\lambda^2 e^{-\lambda}}{2}\right]$$

Or, in the general case:

$$1 - \sum_{k=0}^{N} \frac{\lambda^k e^{-\lambda}}{k!}$$

Applying the Poisson distribution to NGS data

Read_size = 36bp
Mapped_reads = 30,000,000
Alignable_genome_size = 2,700,000,000

$$\lambda = \frac{\text{read\_size} \times \text{mapped\_reads}}{\text{alignable\_genome\_size}}$$

λ = (36 × 30000000)/2700000000 = 0.4

λ: expected read number
k: observed read number

At this λ (mapping coverage), the chance of hitting a tag at any sequence position is, on average, is 4 times in 10

What is the chance hitting 2 tags?

$$P = \frac{\lambda^k e^{-\lambda}}{k!}$$

$$= \frac{0.4^2 e^{-0.4}}{2}$$

= 0.077

- As default MACS defines a peak at $p < 10^{-5}$

Table 1

From the following article

PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls

Joel Rozowsky, Ghia Euskirchen, Raymond K Auerbach, Zhengdong D Zhang, Theodore Gibson, Robert Bjornson, Nicholas Carriero, Michael Snyder & Mark B Gerstein

Nature Biotechnology **27**, 66 - 75 (2009) Published online: 4 January 2009
doi:10.1038/nbt.1518

back to article

Table 1. Genome mappability fraction

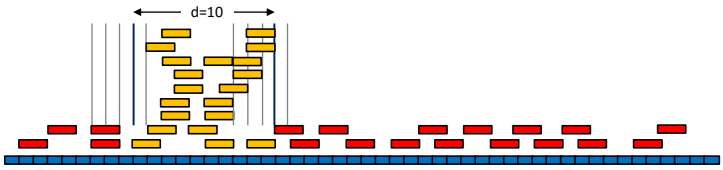
Figures and tables index

		Nonrepetitive sequence		Mappable sequence	
Organism	**Genome size (Mb)**	**Size (Mb)**	**Percentage**	**Size (Mb)**	**Percentage**
Caenorhabditis elegans	100.28	87.01	86.8%	93.26	93.0%
Drosophila melanogaster	168.74	117.45	69.6%	121.40	71.9%
Mus musculus	2,654.91	1,438.61	54.2%	2,150.57	81.0%
Homo sapiens	3,080.44	1,462.69	47.5%	2,451.96	79.6%

For four common model organisms—worm, fruit fly, mouse and human—we have determined the fraction of each genome sequence that is nonrepetitive as well as the fraction that is mappable using 30-nt sequence tags. The genome coverage achievable from genomic tiling arrays corresponds to the nonrepetitive fraction of a genome whereas the mappable coverage is what is achievable by tag-based sequencing approaches. We also determined that as the length of the sequence tags is increased beyond 30, the number of nucleotides in the genomes that are uniquely mappable is 2,452 Mb (79.6%) for 30-nt reads, 2,586 Mb (84.0%) for 40 nt, 2,669 Mb (86.7%) for 50 nt, 2,720 Mb (88.3%) for 60 nt and 2,750 Mb (89.3%) for 70 nt.

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Calculating the significance of a peak

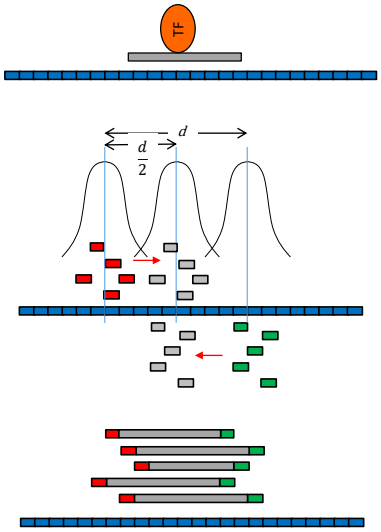


30 tags x 2bp = 60bp
Genome size = 50 bp
λ (chance of a base being covered) = $\frac{60}{50}$ = 1.2
Say d = 10 bp
In 10bp window, k = 4 ($\frac{20\ tags \times 2\ bp}{10\ bp}$)
p = $\frac{\lambda^k e^{-\lambda}}{k!} = \frac{1.2^4 e^{-1.2}}{4\times3\times2\times1}$ = 0.026

This is the p of finding 20 of the 30 tags in a 10bp window

UNIVERSITEIT STELLENBOSCH UNIVERSITY

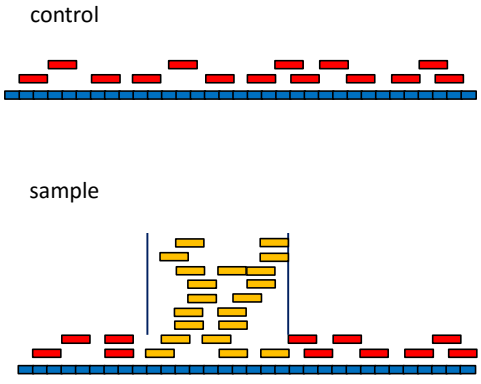
How MACS treat SE and PE samples



- Single end (SE) reads gives Watson and Crick strand reads
- MACS identify W+C peak pairs within a sliding window of 2×sonication size
- *d* is calculated
- Peak center defined at midpoint
- All tags in identified peak shifted by $\frac{d}{2}$ to 3' end

- Paired-end (PE) does not calculate d or shift fragments
- MACS use `--format BAMPE` switch
- Fragment sizes identified by alignment program is used directly

MACS: what the program does



- MACS scales number of tags in sample = number of tags in control
- Scaling can be set to be up (control→tag) or down (tag →control)
- Remove duplicates (PCR bias). No need for *SAMtools > RmDup*
- MACS shift 2*d* window along genome to find significant peaks ($p \leq 10^{-5}$) using λ_{BG} (entire sample)
- The *Summit* of the peak is the position in the identified peak with the highest number of tags
- If there is a control, MACS uses λ_{local} instead of λ_{BG}
- λ_{local} corrects for local chromatin structure, PCR amplification, sequencing bias and genome copy number variation
- MACS calculates $\lambda_{local} = \max(\lambda_{BG}, \lambda_{1k}, \lambda_{5k}, \lambda_{10k})$
- Thus λ is calculated in 1kb, 5kb and 10kb windows centered on identified peaks
- FDR is calculated by identical procedure, but swapping sample and control samples (default *q_fold* < 0.05)

$$Fold\_enrichment = \frac{\text{unit density of tags in the identified peak } (p < 10^{-5})}{\text{unit density of the control } (= \lambda)}$$

BED (Browser Extensible Data) format

Format used by Genome Browsers to display tracks
Text file with tab delimited columns

Required fields:
chrom - The name of the chromosome (e.g. chr3, chrY, chr2_random)
chromStart - The starting position of the feature
chromEnd - The ending position of the feature in the

9 optional fields:
Name, score, strand, thickStart, thickEnd, itemRgb, blockCount, blockSizes, blockStarts

Headers lines provide information and instructions to browser:

```
browser position chr7:127471196-127495720
track name="ItemRGBDemo" description="Item RGB demonstration" visibility=2 itemRgb="On"
chr7    127471196  127472363  Pos1  0  +  127471196  127472363  255,0,0
chr7    127472363  127473530  Pos2  0  +  127472363  127473530  255,0,0
chr7    127473530  127474697  Pos3  0  +  127473530  127474697  255,0,0
```

Files are typically very large
Rather use bigBED format, which only uploads the part of the file that is needed by the browser, to the browser server
You can convert BED -> bigBED files in Galaxy: `Convert Formats > BED-to-bigBED converter`

UNIVERSITEIT STELLENBOSCH UNIVERSITY

bedGraph format

- The bedGraph format allows display of continuous-valued data in track format
- Useful for probability scores and transcriptome data
- The bedGraph format is line-oriented
- Bedgraph data are preceded by a track definition line (no line breaks), which adds a number of options for controlling the default display of this track
- The bedGraph format has four columns of data:

  chrom chromStart chromEnd dataValue

```
browser position chr19:49302001-49304701
browser hide all
browser pack refGene encodeRegions
browser full altGraph
track type=bedGraph name="BedGraph Format" description="BedGraph format" visibility=full color=200,100,0
altColor=0,100,200 priority=20
chr19 49302000 49302300 -1.0
chr19 49302300 49302600 -0.75
chr19 49302600 49302900 -0.50
```

UNIVERSITEIT STELLENBOSCH UNIVERSITY

WIG (wiggle) format

The wiggle (WIG) format is an older format for displaying dense, continuous data such as GC percent, probability scores, etc.
WIG data is lossy: it is compressed and stored internally in 128 unique bins. If you export WIG data from a browser in tabular format, it is not the same as in the original WIG file
Wiggle format is line-oriented.
The first line must be a track definition line (i.e., track type=wiggle_0)with additional options for controlling the display (no line breaks).
WIG data can be `variableStep` or `fixedStep`

```
browser position chr19:49304200-49310700
track type=wiggle_0 name="variableStep" description="variableStep format" visibility=full autoScale=off
viewLimits=0.0:25.0 color=50,150,255 yLineMark=11.76 yLineOnOff=on priority=10
variableStep chrom=chr19 span=150
49304701 10.0
49304901 12.5
49305401 15.0
```

```
fixedStep  chrom=chrN  start=position  step=stepInterval  [span=windowSize]
dataValue1
dataValue2
```

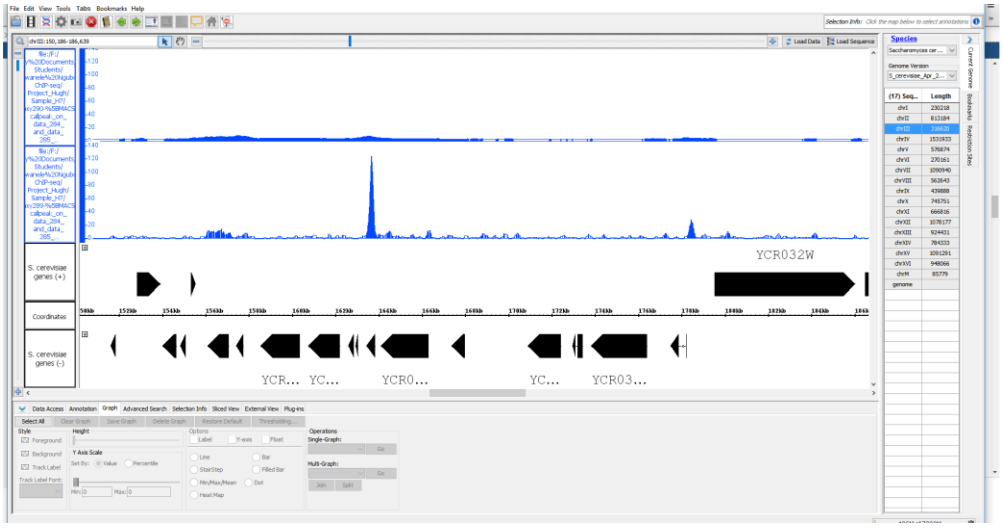
UNIVERSITEIT STELLENBOSCH UNIVERSITY

Program

- Welcome
- Introduction: ChIP-seq, NGS sequencing
- FASTQ file format
- Introduction to Galaxy
  - Registering
  - Get Data to Galaxy
  - Histories
  - Tools Menu
- Quality Control: FastQC, Trim_galore
- Mapping: Bowtie2
- Visualizing mapped sequences: IGB
- Peak calling: MACS
- Visualizing peaks
- Computing on genomic intervals: some tools
- Identifying recognition sequences

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Load bedGraph files into IGB



UNIVERSITEIT STELLENBOSCH UNIVERSITY

Program

- Welcome
- Introduction: ChIP-seq, NGS sequencing
- FASTQ file format
- Introduction to Galaxy
  - Registering
  - Get Data to Galaxy
  - Histories
  - Tools Menu
- Quality Control: FastQC, Trim_galore
- Mapping: Bowtie2
- Visualizing mapped sequences: IGB
- Peak calling: MACS
- Visualizing peaks
- Computing on genomic intervals: some tools
- Identifying recognition sequences

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Select 200bp fragments centered on peaks identified by MACS

- Text manipulation>Compute
- Text manipulation>Cut
- Edit attributes -> Change from “tabular” to “interval”
- Fetch alignments/sequences>Extract genomic DNA
- Motif search>MEME
- MEME currently broken on usegalaxy.org
- See: http://meme-suite.org

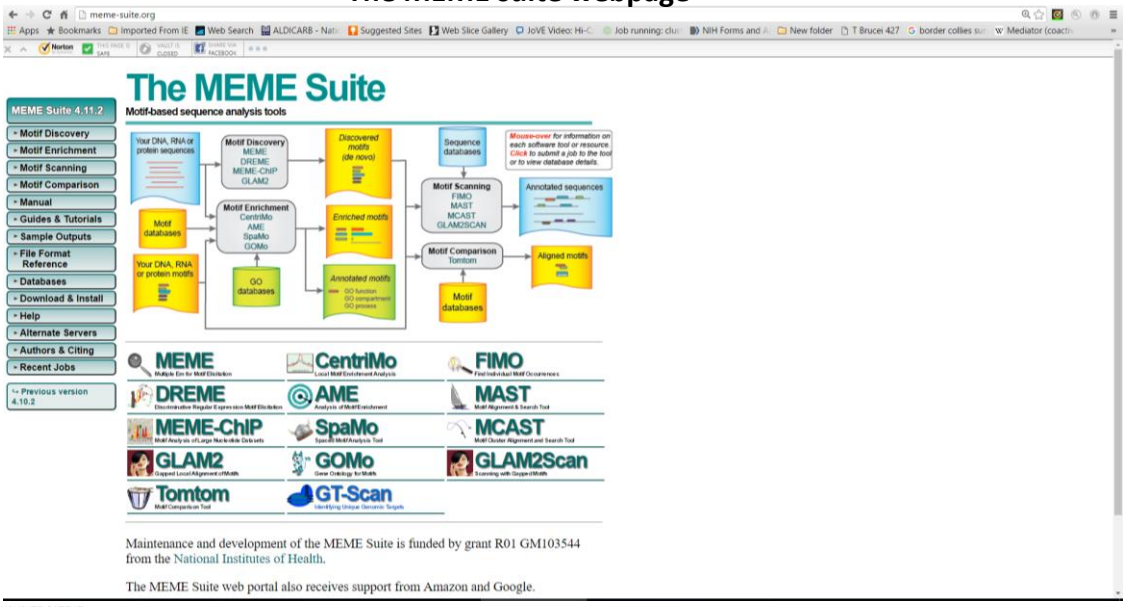
UNIVERSITEIT STELLENBOSCH UNIVERSITY

Program

- Welcome
- Introduction: ChIP-seq, NGS sequencing
- FASTQ file format
- Introduction to Galaxy
  - Registering
  - Get Data to Galaxy
  - Histories
  - Tools Menu
- Quality Control: FastQC, Trim_galore
- Mapping: Bowtie2
- Visualizing mapped sequences: IGB
- Peak calling: MACS
- Visualizing peaks
- Computing on genomic intervals: some tools
- Identifying recognition sequences

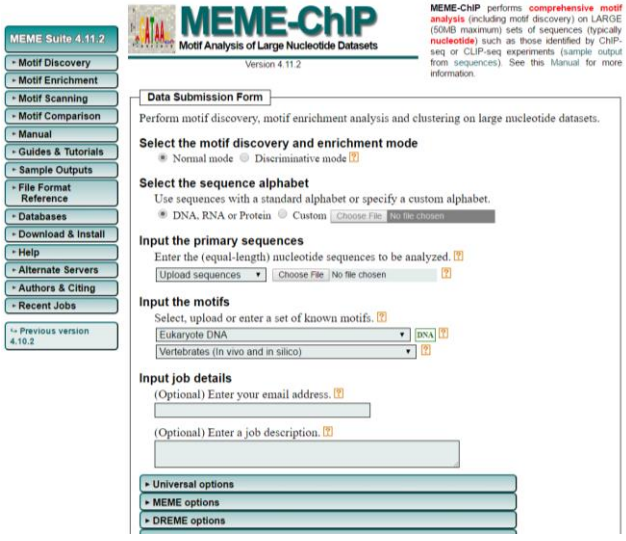
UNIVERSITEIT STELLENBOSCH UNIVERSITY

The MEME suite webpage



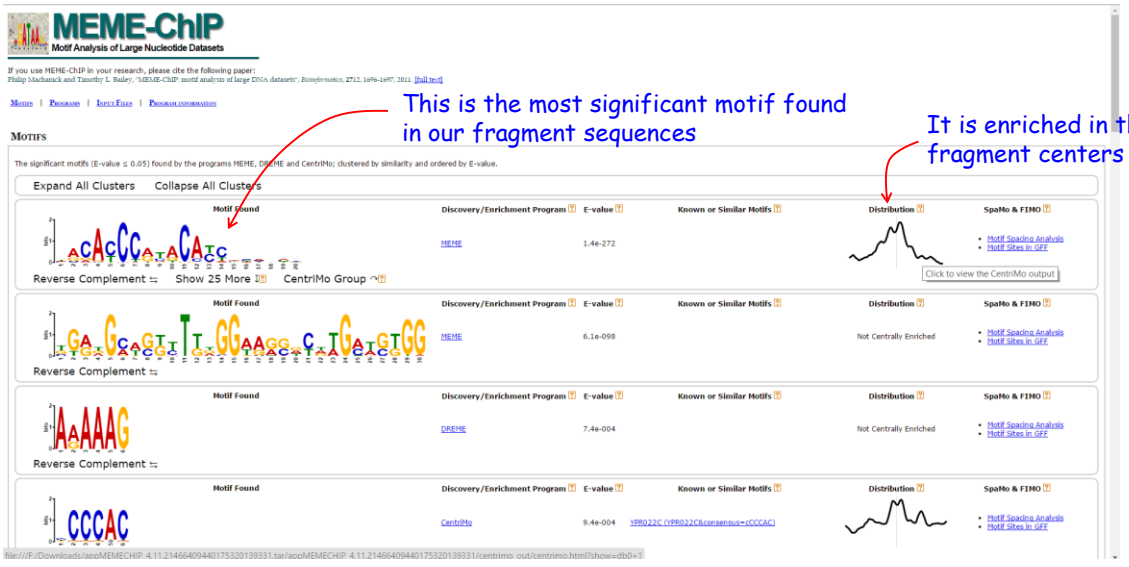
UNIVERSITEIT STELLENBOSCH UNIVERSITY

Upload datafiles to MEME web server



UNIVERSITEIT STELLENBOSCH UNIVERSITY

Output from MEME



Tomtom provides matches to selected database