NovaSeq loading effects on clustering duplicates
Data: multiplexed Chlamydomonas WGS experiment (exp3004, 44 samples)

*Stéphane Plaisance**

January 9th, 2019 - version 1.3

Contents

Software version used for this report:

- BBMap version 38.32 (clumpify)
- bwa version: 0.7.17-r1188
- samtools version 1.9 (htslib)
- Picard version 2.18.17-SNAPSHOT

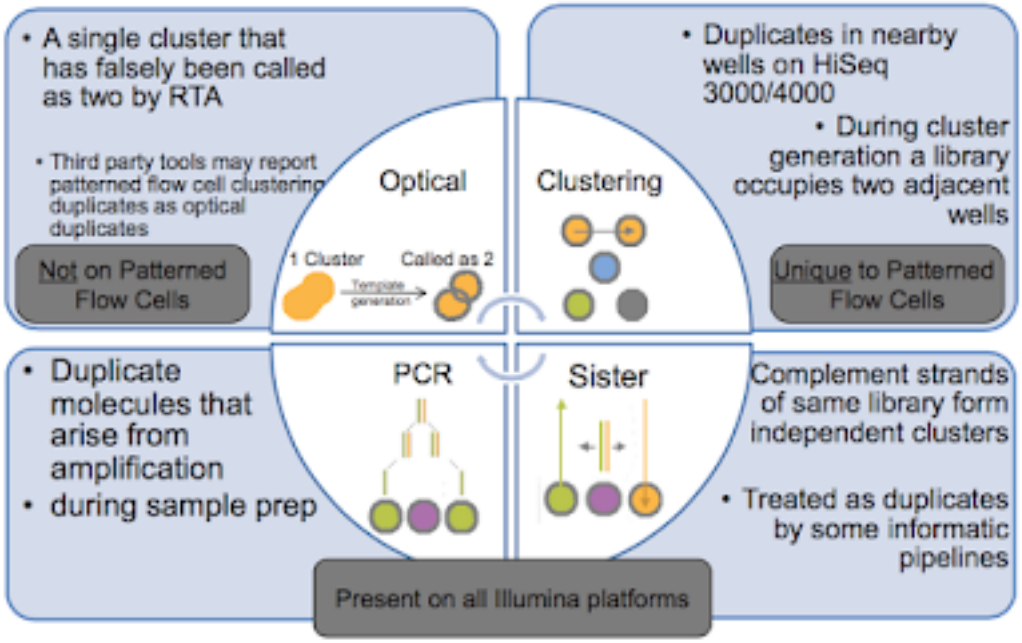
Changes in version 1.2: edited version. version 1.3 with Ilumina feedback

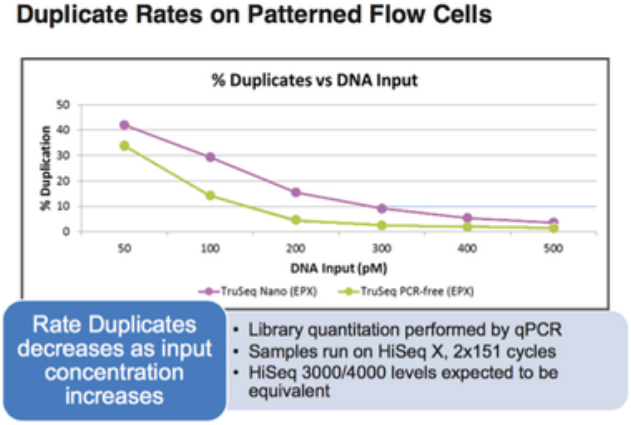
*VIB - Nucleomics Core, nucleomics@vib.be

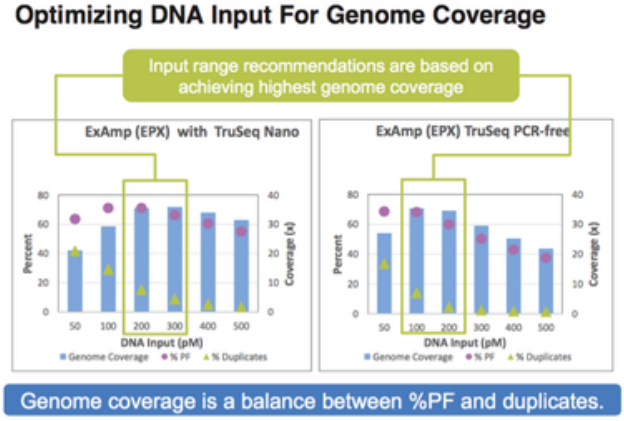
Aim:

A number of reports and discussions suggest that looking for read duplicates may be a good idea when sequence is obtained from the latest Illumina devices. It is however not generally advised to remove the duplicates or at least not for all applications as they may represent true data. Marking duplicates in-placxe is by contrast conservative and will allow troubleshooting in some cases.

Read duplicates may come from different sources as illustrated below, we focus in this work on clustering duplicates as defined on the upper right panel of the first figure as the data comes from a NovasSeq patterned flow-cell.







Source: Illumina

More information

- http://core-genomics.blogspot.com/2016/01/almost-everything-you-wanted-to-know.html
- http://core-genomics.blogspot.com/2016/05/increased-read-duplication-on-patterned.html
- http://seqanswers.com/forums/showthread.php?t=6854

Introduction

A NovaSeq library consisting in a multiplex of 44 separate Chlamydomonas gDNA libraries (PCR-amplified?) was sequenced on three NovaSeq lanes at increasing loading concentrations (fix amount of phi-X).

- Lane2: 1.5nM
- Lane3: 1.75nM
- Lane4: 2nM

After demultiplexing, the paired fastq files obtained for each sample and each lane were processed to identify duplicate read pairs.

Note: The Illumina figures shown above suggest that th e%duplication should decrease with higher loadings.

Read pair duplicates are reads pairs that report the same sequence for both forward and reverse reads respectively. Such reads can result either from fluidics effects during the run or from PCR over-amplification before or during the library preparation. Clustering duplicates (see figure above, top right quadrant) can be identified by their sequence identity AND their vicinity on the flow-cell. PCR duplicates can only be identified by their sequence identity.

Several available tools allow identification and removal of optical/clustering and/or PCR duplicate reads, we use here **BBTools clumpify** as well as **GATK/Picard MarkDuplicates** to compare their performances.

While the first program looks at the read sequence itself using kmer analysis, the second requires that the reads be first aligned to a reference genome as it is based on the comparison of the cigar string present in each mapping row. The clumpify tool runs quite fast and does not require a reference genome while BWA mem (or other mappers) is slow and does rely on a reference. Conversely, clumpify reports/removes only optical/clustering duplicate while Picard details between optical (/clustering) and PCR duplicates.

Results

BBTools clumpify deduplication

The **clumpify** program[1] is used to group similar reads by sequence overlap to apply aggregate calculation but can also be used to identify and optionally remove read duplicates. We applied clumpify to the data using each fastq pair and two different distance cutoff. The shorter distance cutoff (2500) is normally applied to HiSeq data while the larger one (12000) has been defined for NovaSeq patterned cells. We included the analysis results for the lowest concentration on the first lane in order to detect any noticeable difference with the larger 12000 value cutoff.

Statistics and plots for the read-pair counts

The first plot reports the TOTAL number of reads found in each of the 45 sets (44 samples + Undetermined).

`Means per group`

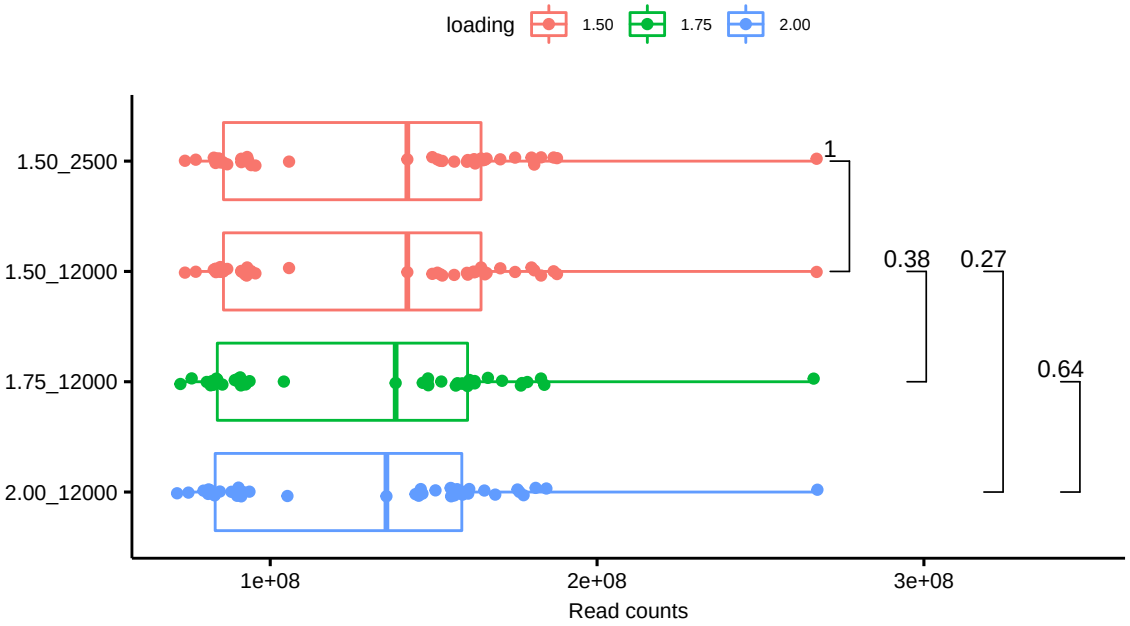
lane	dupd	mean.reads_in	mean.pc_dup
L002	2500	129617864	2.312889
L002	12000	129617864	3.132222
L003	12000	126947290	3.427556
L004	12000	125859266	2.793333

`Statistics (as returned by ggpubr)`

.y.	group1	group2	p	p.adj	p.format	p.signif	method
reads_in	1.50_2500	1.50_12000	1.0000000	1	1.00	ns	Wilcoxon

[1] https://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/clumpify-guide/

.y.	group1	group2	p	p.adj	p.format	p.signif	method
reads_in	1.50_2500	1.75_12000	0.3790775	1	0.38	ns	Wilcoxon
reads_in	1.50_2500	2.00_12000	0.2654428	1	0.27	ns	Wilcoxon
reads_in	1.50_12000	1.75_12000	0.3790775	1	0.38	ns	Wilcoxon
reads_in	1.50_12000	2.00_12000	0.2654428	1	0.27	ns	Wilcoxon
reads_in	1.75_12000	2.00_12000	0.6397541	1	0.64	ns	Wilcoxon



The amount of data obtained with increased loading is not higher than for the initial (default) 1.5nM condition. A slight reduction in read counts may even be observed (ns). The next section evaluates the presence of optical duplicates in the data as a function of flow-cell loading and based on results of the clumpify log results.

Note: The highest points (outliers) of each lane come from the *Undetermined* read group

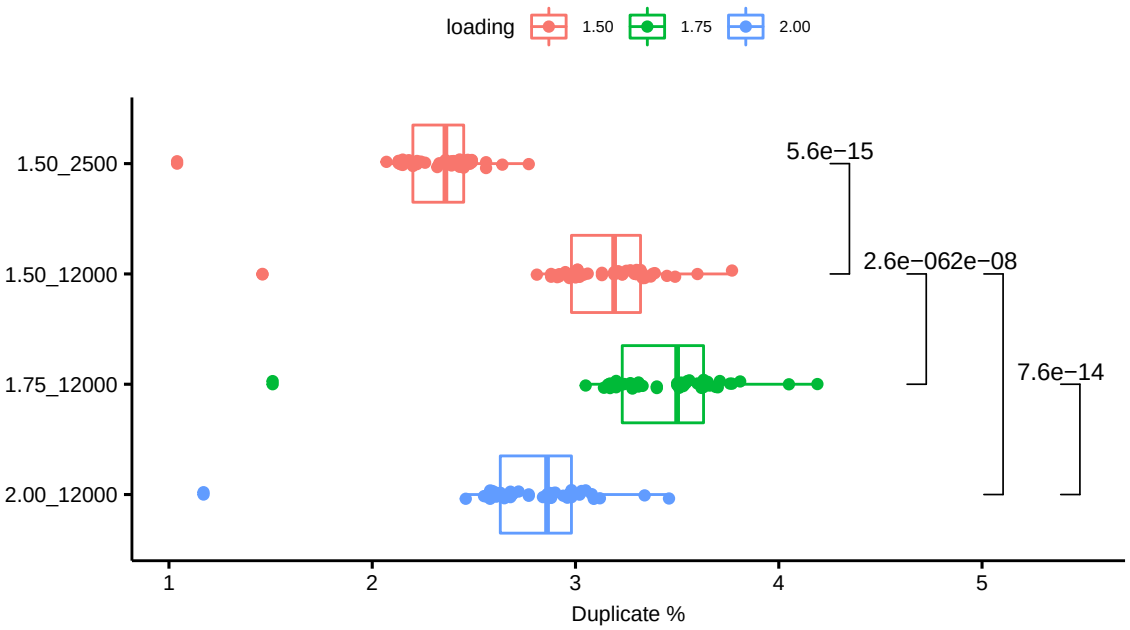
Statistics and plots for the read-pair duplicates

Key groups were compared and tested for their mean difference to evaluate the effect of loading on duplicate counts. Two cutoff values were compared for the first loading to validate the use of the larger one advertised in the clumpify manual for NovaSeq data.

The next plot shows the number of duplicated reads obtained in each lane and for 44 read-pairs + Undetermined.

`Statistics (as returned by ggpubr)`

.y.	group1	group2	p	p.adj	p.format	p.signif	method
pc_dup	1.50_2500	1.50_12000	0.0e+00	0.0e+00	5.6e-15	****	Wilcoxon
pc_dup	1.50_2500	1.75_12000	0.0e+00	0.0e+00	5.6e-15	****	Wilcoxon
pc_dup	1.50_2500	2.00_12000	0.0e+00	0.0e+00	7.3e-14	****	Wilcoxon
pc_dup	1.50_12000	1.75_12000	2.6e-06	2.6e-06	2.6e-06	****	Wilcoxon
pc_dup	1.50_12000	2.00_12000	0.0e+00	0.0e+00	2.0e-08	****	Wilcoxon
pc_dup	1.75_12000	2.00_12000	0.0e+00	0.0e+00	7.6e-14	****	Wilcoxon



Note: The lowest points (outliers) of each lane come from the *Undetermined* read group

As expected, the cutoff value of **12000** recommended by BBtool for NovaSeq data returns significantly more duplicates than the value of **2500** used for HiSeq data (as a longer distance is allowed to define a duplicate pair).

The amount of duplicates does not follow the expected trend based on the increase loading (3.13%, 3.42%, 2.79%). All differences are significant and it is not possible to explain which the highest loading produced less duplicates than the intermediate concentration.

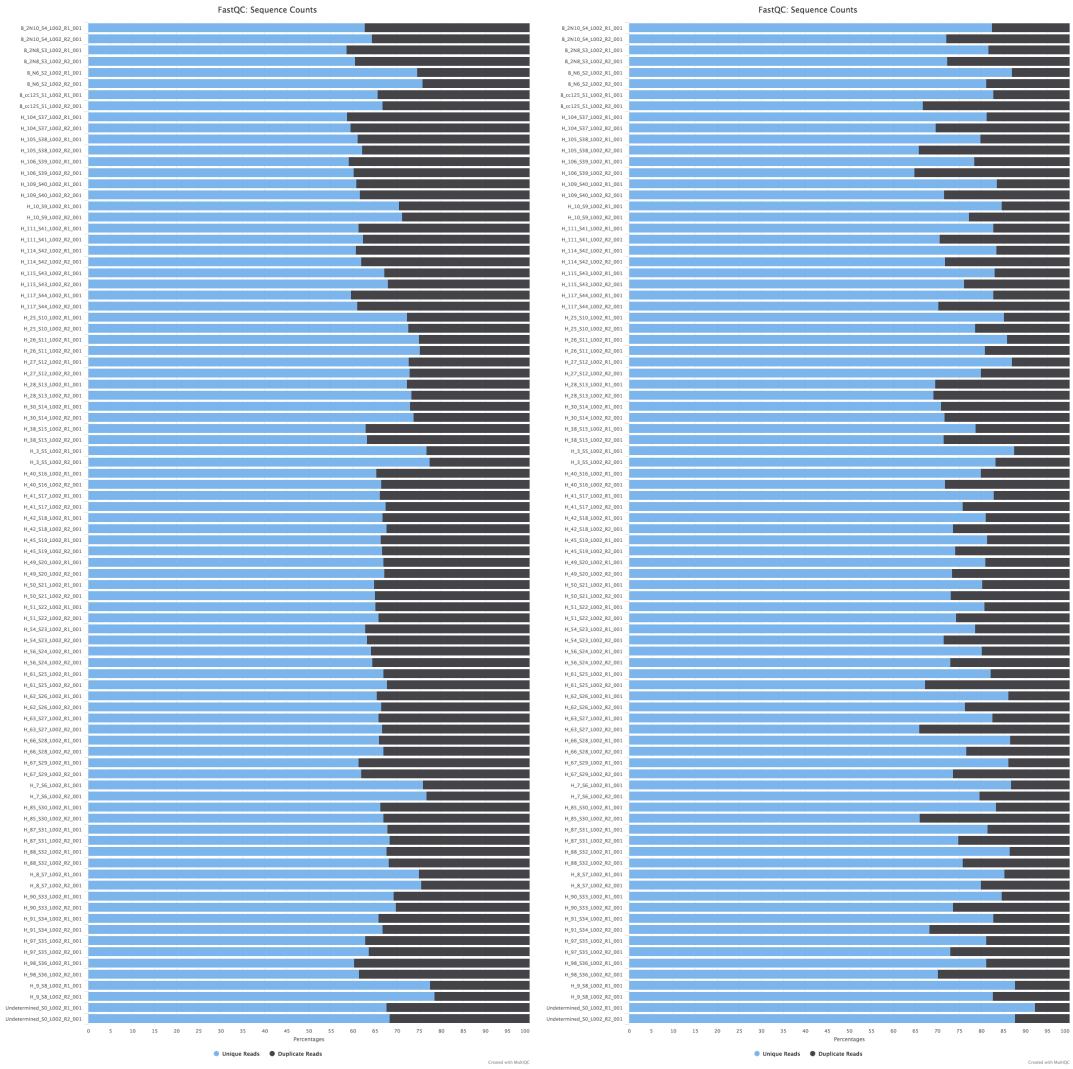
The overall results are however not alarming as **we only found <<5% duplicates** in this experimental data across 44 multiplexed samples (+ Undetermined reads).

FastQC results before and after clumpify

In order to evaluate the effect of clumpify, **FastQC**[2] was run on teh original (raw) reads and on the reads after clupify. **MultiQC**[3] was used to integrate all FastQC results and one plot exported to file.

Note that FastQC only takes the first 100'000 reads to estimate duplication, this could have implications if the first reads correspond to better or worse sequencing events due to the flowcell occupancy.

[2] http://www.bioinformatics.babraham.ac.uk/projects/fastqc/
[3] https://multiqc.info/



FastQC results from the MultiQC report

As seen above, the clumpify processing reduced the amount of duplication, although apparently less for the reverse reads.

Note: *Clumpify supports paired reads, in which case it will clump based on read 1 only.*

This suggests that there might be duplicates in the second read file of each pair that were not removed by clumpify.

Second round of deduplication with clumpify

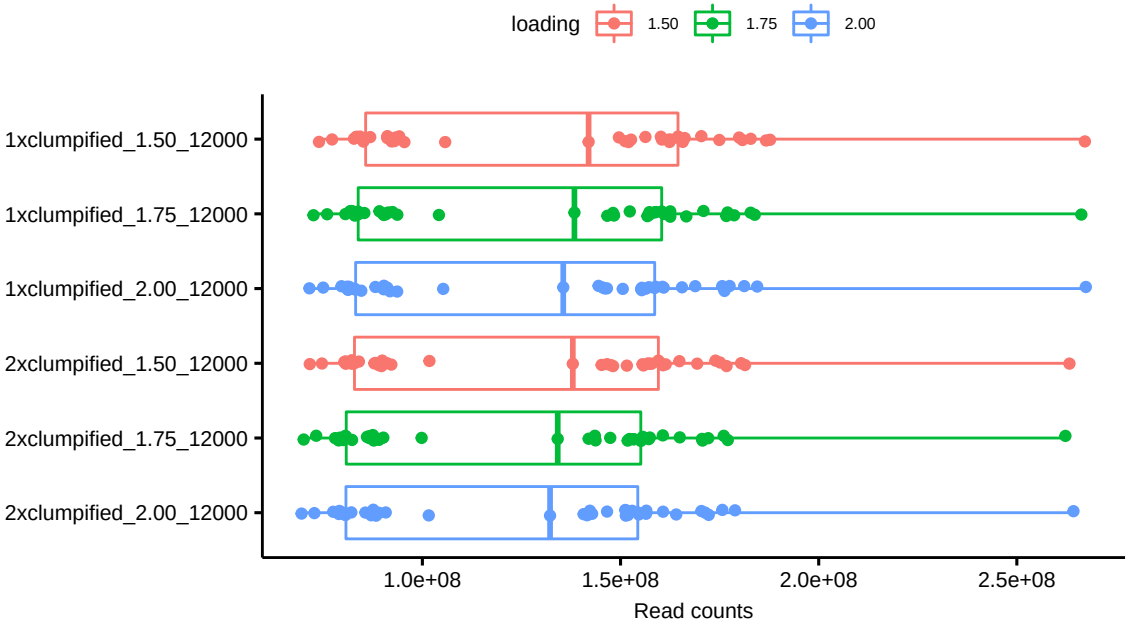
To control for the presence of leftover duplicates in the clummpified R2 files suggested by the clumpify manpage, the reads from all clumpified sample above where reverse-complemented and the new FastQ pairs subjected to a new round of **clumpify** deduplication (R1=>rc=>R2'; R2=>rc=>R1').

`Means per group`

group	mean.reads_in	mean.pc_dup
1xclumpified_1.50_12000	129617864	3.1322222

group	mean.reads_in	mean.pc_dup
1xclumpified_1.75_12000	126947290	3.4275556
1xclumpified_2.00_12000	125859266	2.7933333
2xclumpified_1.50_12000	125633999	0.1177778
2xclumpified_1.75_12000	122684403	0.1408889
2xclumpified_2.00_12000	122419812	0.1180000

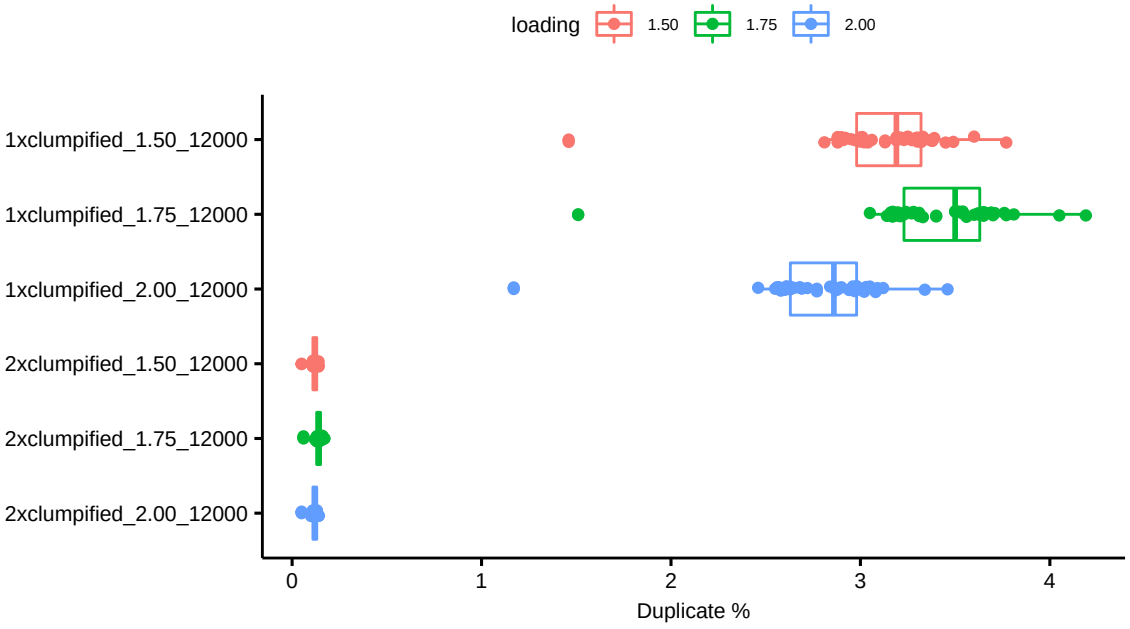
The first plot reports the number of reads found in each of the 45 sets before and after the second round of clumpify (44 samples + Undetermined).“)



Note: The highest points (outliers) of each lane come from the *Undetermined* read group

The new fastq files contain less reads but the trends are identical between loadings (stats=ns).

The next plot shows the number of duplicated reads obtained in each lane and for 45 read-pairs before and after the secound round of clumpify.



Note: The lowest points (outliers) of each lane come from the *Undetermined* read group

A low level of read duplicates were found in this read pair (**less than 0.2%**), thereby confirming that the clumpify de-duplication was almost complete after 1 pass. The trends of duplication remain identical across loadings

Mapping paired reads to the reference genome to identify duplicates

Other tools are available to detect duplicates but they rely on mappings instead of raw reads. In order to appply these tools, we therefore started by aligning the read data from a selected sample to the Chlamydomonas reference genome.

A publicly available reference genome for *Chlamydomonas reinhardtii* (v5.5) was published (Merchant et al. 2007) and is available at **phytosome** after registering[4][5] and was used for this purpose.

Mapping the reads to the Chlamydomonas genome

The main genome assembly is approximately 111.1 Mb arranged on 17 chromosomes and 37 minor scaffolds (numbered 18 to 54)

Because BWA mapping is quite slow, only reads from sample **B_cc125_S1** were mapped to the reference using BWA mem (Li and Durbin 2010)

Mapping derived duplicates counts using samtools mrkdup

Samtools metrics are summarized below and reflect the BWA results as analysed by **samtools flagstat** (Li et al. 2009)

Zero duplicates found! : BWA doesn't mark duplicates and Samtools flagstat will only returns duplicate counts in the BAM file if they are marked as duplicates after BWA mapping.

In order to include samtools PCR duplicate counts in the flagstat report, the BWA mappings needed to be post-processed in several steps using other samtools commands (sort, fixmate, and finally mrkdup).[6] The final step *samtools mrkdup -s* searches for PCR duplicates based on the read sequence and returns duplicate counts for paired-reads, single reads, and total-reads. The percentage of duplicates were computed from the samtools counts with respect to the total number of examined reads in the same category (single/paired/total; excluded reads are not counted in teh ratios).

`Samtools mrkdup metrics for B_cc125_S1`

	7	8	9
sample	B_cc125_S1_L002	B_cc125_S1_L003	B_cc125_S1_L004
READ	83769039	82551060	82048812
WRITTEN	83769039	82551060	82048812
EXCLUDED	15197996	15009632	14876873
EXAMINED	68571043	67541428	67171939
PAIRED	66426668	65446802	65109036
SINGLE	2144375	2094626	2062903
DUPLICATE_PAIR	6656222	6029932	5352856
DUPLICATE_SINGLE	1307802	1270916	1244755
DUPLICATE_TOTAL	7964024	7300848	6597611
single_pc	3.127231	3.101246	3.071079
single.dup_pc	60.98756	60.67508	60.33997
paired.dup_pc	10.020406	9.213486	8.221372
dup_pc	9.507121	8.844039	8.041080

[4]https://phytozome.jgi.doe.gov/pz/portal.html#!info?alias=Org_Creinhardtii
[5]https://genome.jgi.doe.gov/portal/pages/dynamicOrganismDownload.jsf?organism=Creinhardtii
[6]http://www.htslib.org/doc/samtools.html

Note: although few examined reads are found unpaired (single; ~3%) duplicates within these reads are very frequent (~60%) when comparing to the examined number of single reads.

Mapping metrics with samtools flagstat

In order to obtain mapping information, Samtools flagstat was also applied to the enriched BAM data and results reported next.

`Samtools flagstat metrics for B_cc125_S1`

	7	8	9
sample	B_cc125_S1_L002	B_cc125_S1_L003	B_cc125_S1_L004
in_total	83769039	82551060	82048812
secondary	367951	359630	353628
supplementary	0	0	0
duplicates	7964024	7300848	6597611
mapped	68938994	67901058	67525567
paired_in_sequencing	83401088	82191430	81695184
read1	41700544	41095715	40847592
read2	41700544	41095715	40847592
properly_paired	62700380	61854770	61628302
with_itself_and_mate_mapped	66426668	65446802	65109036
singletons	2144375	2094626	2062903
with_mate_mapped_to_a_different_chr	3357430	3231360	3126794
with_mate_mapped_to_a_different_chr_(mapQ>=5)	2734410	2628896	2540072
mapped_pc	82.29651	82.25341	82.29926
paired_pc	99.56076	99.56435	99.56900
dup_pc	9.507121	8.844039	8.041080

There is no dramatic difference between lanes from the flagstat values obtained after mapping this sample to the reference genome.

Finding duplicates with Picard MarkDuplicates

Picard MarkDuplicates (Broad Institute (Accessed: 2018/07/23)) detect both sequencing duplicates and optical duplicates.

Picard MarkDuplicates was run with *OPTICAL_DUPLICATE_PIXEL_DISTANCE* set to 2500 as read from the manpage and the different reported metrics are detailed in the next table.

```
OPTICAL_DUPLICATE_PIXEL_DISTANCE=Integer
                        The maximum offset between two duplicate clusters in order to consider them optical
                        duplicates. The default is appropriate for unpatterned versions of the Illumina platform.
                        For the patterned flowcell models, 2500 is moreappropriate. For other platforms and
                        models, users should experiment to find what works best.  Default value: 100. This option
                        can be set to 'null' to clear the default value.
```

`Picard MarkDuplicates metrics for B_cc125_S1`

	7	8	9
SAMPLE	B_cc125_S1_L002	B_cc125_S1_L003	B_cc125_S1_L004
LIBRARY	lib-B_cc125_S1	lib-B_cc125_S1	lib-B_cc125_S1
UNPAIRED_READS_EXAMINED	2144375	2094626	2062903
READ_PAIRS_EXAMINED	33213334	32723401	32554518
SECONDARY_OR_SUPPLEMENTARY_RDS	367951	359630	353628
UNMAPPED_READS	14830045	14650002	14523245
UNPAIRED_READ_DUPLICATES	1307802	1270916	1244755
READ_PAIR_DUPLICATES	3332992	3019549	2680962
READ_PAIR_OPTICAL_DUPLICATES	946718	1247446	1075048
PERCENT_DUPLICATION	0.116285	0.108230	0.098355

	7	8	9
ESTIMATED_LIBRARY_SIZE	207257051	268943219	297948767
pc_unpair	6.456368	6.401003	6.336764
pc_unpairdup	60.98756	60.67508	60.33997
pc_pairdup	10.035102	9.227491	8.235299
pc_optdup	2.850415	3.812092	3.302300

Note: Although only 6% of the reads are found unpaired, 60% thereof ar marked as duplicates and may constitute a set of artefactual reads. It may be wise, for wome applications, to ignore unpaired reads in later analyses.

Picard Metrics definitions:

LIBRARY: The library on which the duplicate marking was performed.

UNPAIRED_READS_EXAMINED: The number of mapped reads examined which did not have a mapped mate pair, either because the read is unpaired, or the read is paired to an unmapped mate.

READ_PAIRS_EXAMINED: The number of mapped read pairs examined. (Primary, non-supplemental)

SECONDARY_OR_SUPPLEMENTARY_RDS: The number of reads that were either secondary or supplementary

UNMAPPED_READS: The total number of unmapped reads examined. (Primary, non-supplemental)

UNPAIRED_READ_DUPLICATES: The number of fragments that were marked as duplicates.

READ_PAIR_DUPLICATES: The number of read pairs that were marked as duplicates.

READ_PAIR_OPTICAL_DUPLICATES: The number of read pairs duplicates that were caused by optical duplication. Value is always < READ_PAIR_DUPLICATES, which counts all duplicates regardless of source.

PERCENT_DUPLICATION: The fraction of mapped sequence that is marked as duplicate.

ESTIMATED_LIBRARY_SIZE: The estimated number of unique molecules in the library based on PE duplication.

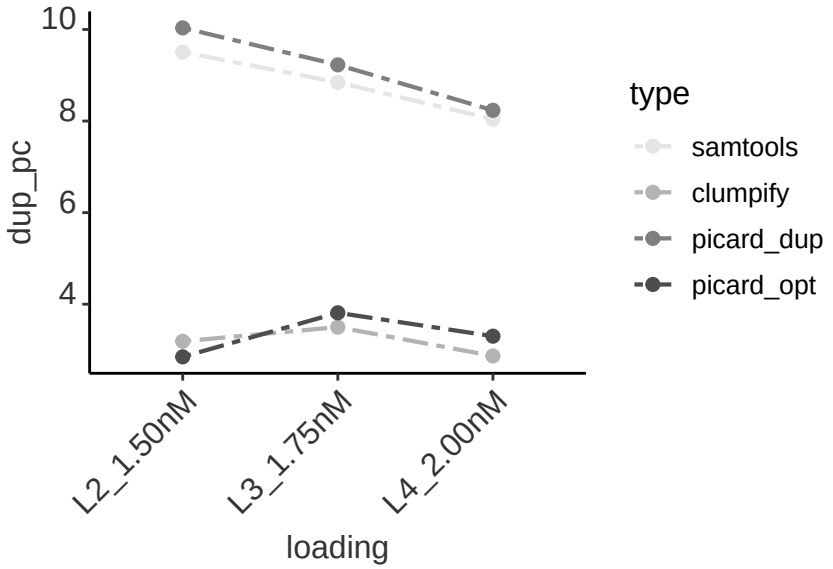
Note: Picard look for duplicates only in the mapped reads while clumpify processes all read pairs.

Comparing duplicate trends between the three methods and across loadings

The next plot shows samtools, clumpify, and Picard MarkDuplicates results side by side with different trends (lines) across loadings.

`Duplicate counts for B_cc125_S1 from the three methods`

	L2_1.50nM	L3_1.75nM	L4_2.00nM
samtools	9.51	8.84	8.04
clumpify	3.19	3.50	2.87
picard_dup	10.04	9.23	8.24
picard_opt	2.85	3.81	3.30



Both Picard and clumpify report more "optical" duplicates for the middle concentration.

Discussion and Conclusion

Both **BBTools clumpify** and **Picard MarkDuplicates** report a similar fraction of optical/clustering duplicates which suggests that the clumpify method (faster and not requiring mapping) can be used. Clumpify is efficient after a single deduplication round and processing on the reverse complemented pairs is not necessary when deduplication would be applied.

Samtools mrkdup only marks and/or remove sequence-based duplicates which include potential PCR duplicates but it does not use the flowcell coordinates to identify the optical/clustering subset. The total duplicate counts reported by **Samtools mrkdup** match the values obtained with **Picard MarkDuplicates** for the same category, cross validating both tools.

It is not clear why increasing loading concentrations led to a biphasic duplicate trend for the clustering/optical class of duplicates (up, then down). It is possible that 1.75 corresponds to a optimal loading in this experiment but this is purely hypothetical and not reflected by the trend for total duplicates nor by the total amount of reads across loading concentrations.

The global results obtained here do not suggest that 2nM is excessive based on duplicate counts. However, as expected from the first Illumina figure, higher loading produced slightly less reads than the standard 1.5nM concentration (but statistical test returned not significant).

The full mapping of all samples being time consuming, only one sample was used for the Picard test. **Picard MarkDuplicates** was used with a pixel distance of 2500, similar to the value used for Hiseq data. As read below, it seems that Picard does not require this value to change for NovaSeq data as opposed to clumpify (asked but not confirmed by the Picard developers).

The results of this experiment suggest that increasing the loading from 1.5nM to 2nM is not affecting duplicate yields but does not produce more data either (even slightly less). Based on this experiment, this work suggests to stick to the recommendations and load sequencing libraries at up to 1.5nM on NovaSeq flow-cells.

Review and feedback from Illumina

(from dschaerlaekens@illumina.com; Jan-04-2019)

Dear Kizi, Dear Stephane

Thank you for providing me with the SAV data of the NVSQ XP run of 13-DEC-2018 and the detailed document regarding the loading effects on the clustering duplicates.

The run had a Q30 of 89.4%, generated 11,4 Billion reads PF and exceeds the specifications as listed on this webpage: https://emea.illumina.com/systems/sequencing-platforms/novaseq/specifications.html

On lane 1 we loaded 1.5 nM of a SMART-Seq NEBNext library (Exp2865). On lane 2-4 we loaded different concentrations (1.5; 1.75; 2.0 nM) of a NEBNext library (Exp3004).

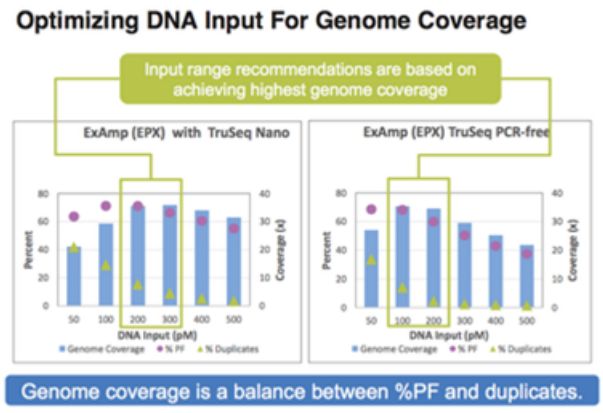
Please have a look to this overview with some crucial parameters:

LANE	LIBRARY	CONC (nM)	%PF	%Occ	phix	samtools	clumpify	Picard_dup	Picard_opt
1	Exp2865	1.5	74.07	84	1.26				
2	Exp3004	1.5	76.15	94	0.82	9.51	3.19	10.04	2.85
3	Exp3004	1.75	74.58	94.3	0.74	8.84	3.5	9.23	3.81
4	Exp3004	2.0	73.94	94.8	0.66	8.04	2.87	8.24	3.30

These are some comments: In general %PF of all lanes is very good, as well as the low % total duplicates and low % clustering duplicates. If further optimization is possible, it will be about small effects only.

The %PF of lane 1 (Exp2865) is 74.07% while the % occupancy is 84% (with phix >1%). This means that 25.03% of the wells are not passing filter, and 16% are not passing filter because they are empty. If you want to optimize, you could test a loading concentration that is just a little bit higher (e.g. 1.6 nM), in order to fill more wells and boost the %PF with a few %.

The % occupancy of lane 2 is much higher (94%; and phix <1%). 23.85% of the wells are not passing filter, but only 6% is not passing filter because they are empty. The other 17.85% is not passing filter due to polyclonality, showing that it would not be a good idea to load higher concentrations. Also the low % duplicates observed for these lanes show that you are on the right side in this graph:



Source: Illumina

If you want to further optimize, you could try to decrease the loading concentration a little bit (e.g. to 1.4 nM) and test if the % duplicates stays within acceptable limits while increasing the % of monoclonal clusters PF with a few %.

Please tell me if you have any further questions.

With kind regards Dirk Schaerlaekens

– Dirk Schaerlaekens Field Applications Scientist Illumina Website: www.illumina.com Email: dschaerlaekens@illumina.com

Technical Support Belgium (toll free): 0800 77160 Netherlands (toll free): 0800 0222 493 All other European time zones: +44 1799 534000 http://www.illumina.com/support.ilmn techsupport@illumina.com

Technical Bulletins: https://support.illumina.com/bulletins.html Trainings: http://support.illumina.com/training.ilmn

last edits: 2019/01/09

References

Broad Institute. (Accessed: 2018/07/23). "Picard Tools; Version 2.18.11-Snapshot." *Broad Institute, GitHub repository.* http://broadinstitute.github.io/picard/.

Li, H., and R. Durbin. 2010. "Fast and accurate long-read alignment with Burrows-Wheeler transform." *Bioinformatics* 26 (5): 589–95.

Li, H., B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, and R. Durbin. 2009. "The Sequence Alignment/Map format and SAMtools." *Bioinformatics* 25 (16): 2078–9.

Merchant, Sabeeha S, Simon E Prochnik, Olivier Vallon, Elizabeth H Harris, Steven J Karpowicz, George B Witman, Astrid Terry, et al. 2007. "The Chlamydomonas Genome Reveals the Evolution of Key Animal and Plant Functions." *Science (New York, N.Y.)* 318 (5848): 245–50. doi:10.1126/science.1143609.