

Exome-sequencing quality control with TEQC

Oliver Hofmann

July 12, 2011

A sample data quality analysis of an exome sequencing data set. Sequences analyzed are identical to those used in the lab session and consist of data for chromosome 22 only; keep this in mind when looking at information such as genomic coverage.

Let's start by looking at the target regions the experiment meant to capture. Those are stored as `textttBED` files, but can be any format:

```
> library("TEQC")

> targets <- get.targets(targetsfile = "../Visualization/hg19_targets.bed",
+   chrcol = 1, startcol = 2, endcol = 3, zerobased = F,
+   skip = 1)

[1] "read 441 (non-overlapping) target regions"

> ft <- fraction.target(targets, genome = "hg19")
> ft

[1] 2.371507e-05
```

That's a rather small fraction of the overall genome. Let's compare this to the sequencing reads that aligned to *chr22*. Reads were generated from the BAM file using `bamToBed`, a part of the `bedTools` package.

```
> reads <- get.reads("../Intermediates/galaxy.bed",
+   chrcol = 1, startcol = 2, endcol = 3, idcol = 4,
+   zerobased = F, skip = 0)

[1] "read 511067 sequenced reads"

> fraction.reads.target(reads, targets)

[1] 0.4941661
```

Almost half of our reads align to the target regions; given that they are only a very small percentage of the overall genome this is quite good. Allowing for a 100bp border around the target region increases the coverage even further:

```
> fr <- fraction.reads.target(reads, targets, Offset = 100)
> fr

[1] 0.5869035
```

All kinds of additional information is available, down to detailed coverage data for each single target:

```
> cov <- coverage.target(reads, targets, perTarget = T,
+   perBase = T)
> cov
```

\$avgTargetCoverage
[1] 184.4568

\$targetCoverageSD
[1] 129.0718

\$targetCoverageQuantiles
0% 25% 50% 75% 100%
0 80 176 271 653

\$targetCoverages
RangedData with 441 rows and 2 value columns across 1 space

	space	ranges	avgCoverage
	<factor>	<IRanges>	<numeric>
1	chr22	[17618910, 17619247]	279.9556
2	chr22	[17619439, 17619628]	236.5053
3	chr22	[17621948, 17622123]	140.6705
4	chr22	[17622282, 17622467]	287.9247
5	chr22	[17623987, 17624021]	205.6571
6	chr22	[17625913, 17626007]	239.3053
7	chr22	[17629337, 17629450]	229.6842
8	chr22	[17630431, 17630635]	170.4195
9	chr22	[17640015, 17640169]	0.0000
...
433	chr22	[51019848, 51019982]	246.84444
434	chr22	[51020177, 51021394]	74.00575
435	chr22	[51063572, 51063892]	106.60748
436	chr22	[51064006, 51064109]	152.96154
437	chr22	[51064363, 51064491]	146.59690
438	chr22	[51064581, 51064706]	124.42857
439	chr22	[51065018, 51065188]	153.91813
440	chr22	[51065261, 51065834]	105.46341
441	chr22	[51065983, 51066201]	29.43836

coverageSD
<numeric>

1	84.01972
2	81.09455
3	46.03209
4	63.76091
5	24.03544
6	46.33804
7	28.09413
8	49.05399

```

9      0.00000
...    ...
433    30.63893
434    96.15896
435    75.08621
436    25.80883
437    37.90439
438    30.32238
439    72.23519
440    96.12048
441    12.65272

$coverageAll
SimpleRleList of length 1
$chr22
'integer' Rle of length 51244519 with 438546 runs
  Lengths: 16960311      21      50 ...      25      46
  Values :          0          1          2 ...          2          1

$coverageTarget
SimpleRleList of length 1
$chr22
'integer' Rle of length 74398 with 55870 runs
  Lengths:  1  1  1  2  1  1 ...  4  9  1  6  3
  Values : 180 181 186 191 192 194 ...  8  7  6  7  6

```

Overall, more than 96% of target regions are covered, and more than 90% of target nucleotides exceed ten-fold coverage (also see figure 1). We can also look at target coverage uniformity (figure 2). This is independent of the total number of reads and helps comparing the run quality between samples.

```

> covered.k(cov$coverageTarget, k = c(1, 5, 10))

          1          5          10
0.9639910 0.9310869 0.9126455

```

Additional information is available and can be queried. For example, we can explore whether there is any data dependency on the size of the target region. This will depend on the *bait tiling*, that is, the correlation between the regions we are interested in, and the hybridization capture probes. For this we need to extract the number of reads overlapping each target which we can then use to create different views such as in figure 3.

```

> targets2 <- cov$targetCoverages
> targets2 <- readsPerTarget(reads, targets2)
> coverage.targetlength.plot(targets2, plotcolumn = "nReads",
+   pch = 16, cex = 1.5)

> coverage.targetlength.plot(targets2, plotcolumn = "avgCoverage",
+   pch = 16, cex = 1.5)

```

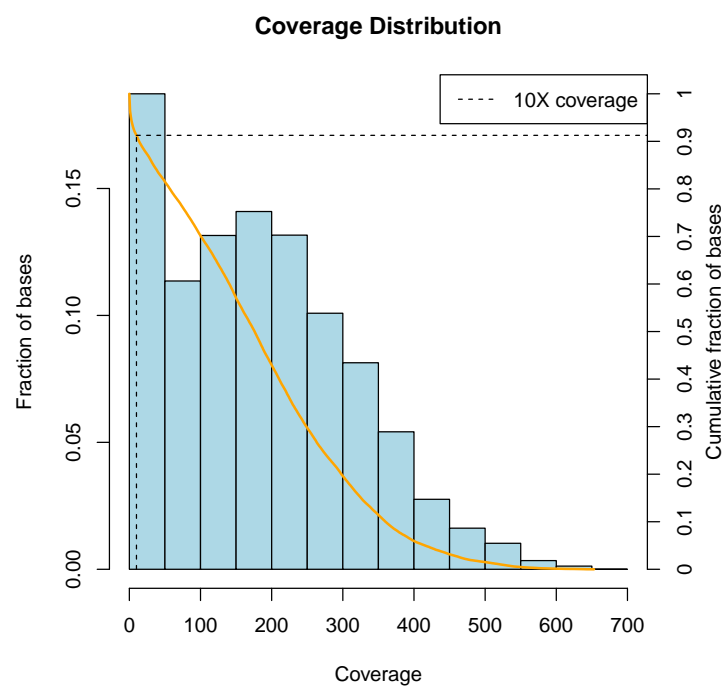


Figure 1: Histogram and cumulative density of target base coverage

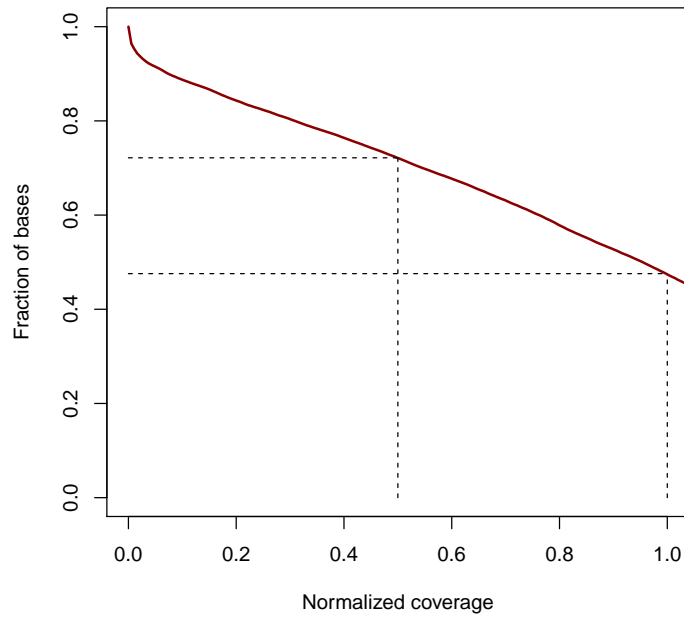


Figure 2: Cumulative fraction of target bases that reach at least a given normalized coverage (per-base coverage divided by the average coverage over all target bases)

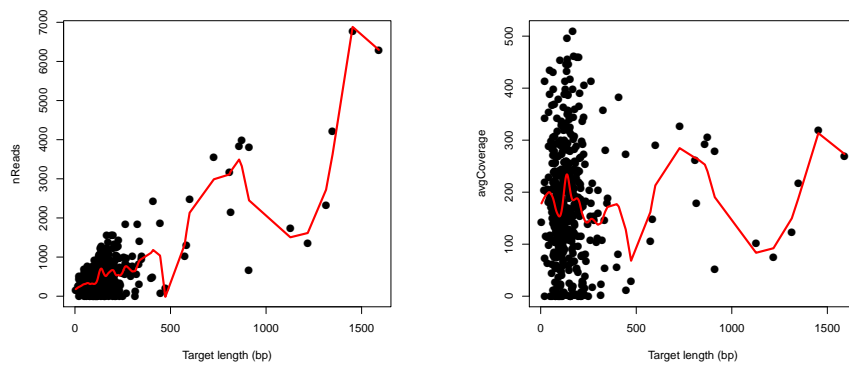


Figure 3: Number of reads and average coverage depending on target size

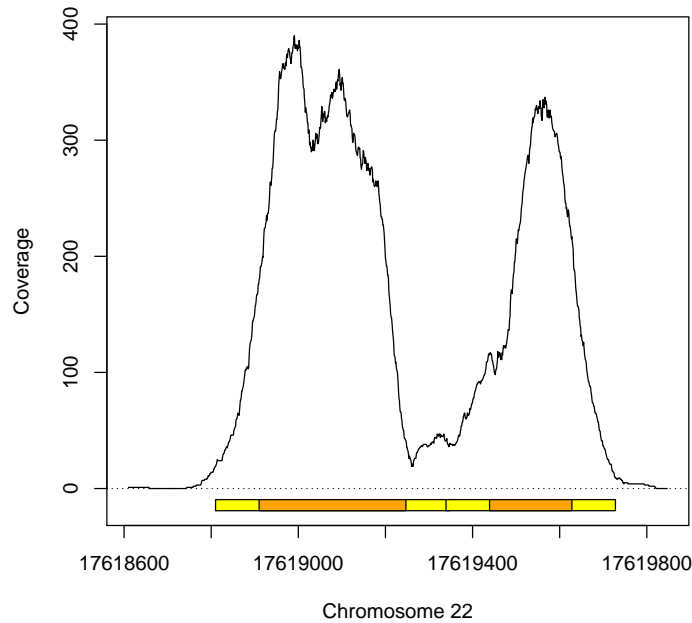


Figure 4: Per-base coverage of the target region between nucleotides 17618910 and 17619247 (orange) on *chr22* plus 100bp on each side (yellow)

The same target coverage information can be used to look at individual target regions in more detail, for example to assess the uniformity of coverage (see figure 4 for the output):

```
> coverage.plot(cov$coverageAll, targets, Offset = 100,
+   chr = "chr22", Start = 17618610, End = 17619847)
```

Another quality control measure is the impact of read duplicates. This can either be a result of extremely high coverage or, more frequently, originate from PCR artifacts. Unlike in ChIP-seq and WGS analysis here we do expect a fair amount of duplication due to the strong target enrichment. Figure 5 shows that for our single-end read data a large proportion have multiple copies 'on target', indicating real duplication rather than artifacts. At the same time the majority of unique reads are off target. However, removing such singleton reads also removes the largest individual group of reads that are 'on target' and needs to be performed with caution or in combination with other quality filters.

While it is usually advisable to remove duplicates, or at least to collapse them into a single read, this will significantly reduce overall target coverage depth with subsequent impact on variant calls.

The final quality control check is one of reproducibility, either across technical or biological replicates. For the purpose of this document I've merely subsampled the first data set randomly and compared it to the original. In

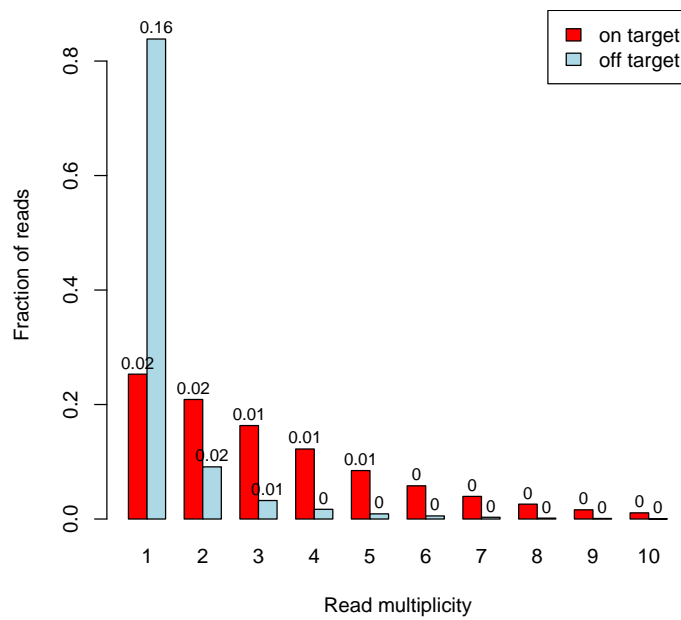


Figure 5: Duplicate barplot showing fractions of on- and off-target reads that are unique, present twice, three times, etc. (x-axis). Numbers on top of the bars represent absolute read counts in millions; read duplication above 10 copies not shown.

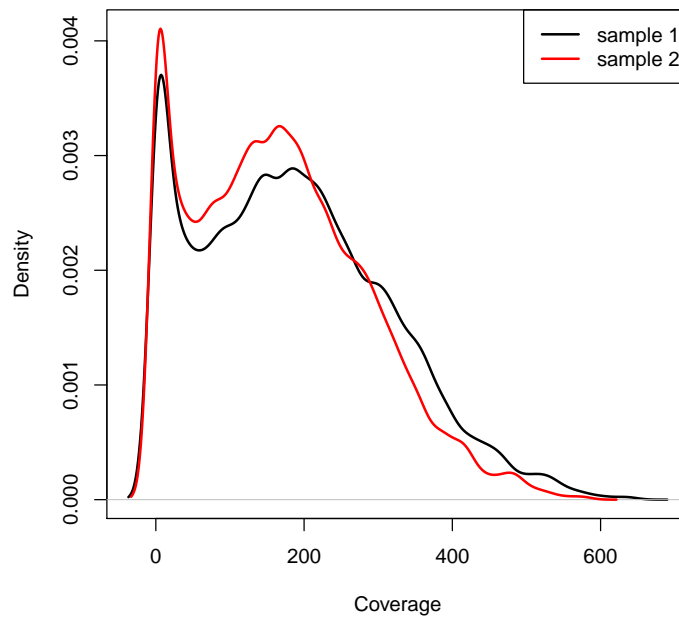


Figure 6: Non-normalized target coverage densities of two samples

practice you'd compare different data sets to identify batch problems, for example by looking at differences in coverage (figure 6) or correlation structure (figure 7).

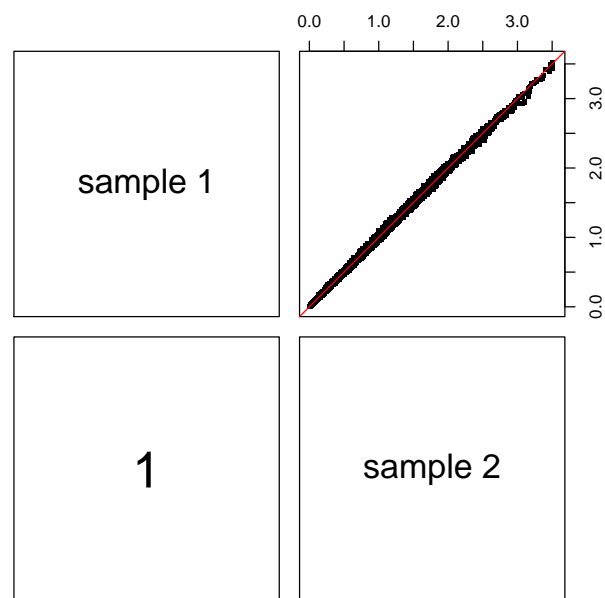


Figure 7: Correlation of replicate sample target coverage