BCBIO 401 RNA-Seq Homework Dylan Khor

1ai) The dataset I used is Nod1_1_S61_L001_R1_001.fastq.gz and the adapter I used is Illumina adapter 1, which has the sequence AGATCGGAAGAGCACACGTCTGAACTCCAGTCA.

1aii) 3068957 reads matched the adapter exactly.

1aiii) Almost all of the adapter sequence appears near the end of the reads but none appear at the start of the reads, which shows some evidence of nonrandomness. Also, I expect the whole adapter sequence to not be present in the reads, however, as evidenced by my data, a lot of the reads have the whole adapter sequence present in it. The likely reason for this is that there are lots of short cDNA fragments. Other than that, there appears to be two reads with the exact same sequence:

```
TTAATAGATTTGGGTCGCATTGACCCCAGTCAACCCATCGATTTAACTCAGCTTGAGATCGGAAGAGCACACGTCTGAACTCCAGTCACACGAATCCATCT
TTAATAGATTTGGGTCGCATTGACCCCAGTCAACCCATCGATTTAACTCAGCTTGAGATCGGAAGAGCACACGTCTGAACTCCAGTCACACGAATCCATCT
```

This indicates a strong indication of nonrandomness, because the chances of two reads this long being same is extremely low. The possible reason for this is due to optical duplication, which happens when two pair of reads are both on the same tile in the flow cell, and they are overlapping with each other.

AdapterRemoval found 14677234 matches for reads with the adapter sequence specified in (1ai). This is because AdapterRemoval not only finds reads with exact matches to the complete adapter sequence, but it also looks for reads that contains readthrough of the adapter sequence, which is why AdapterRemoval found more matches for reads with adapter sequence than zgrep, which only consider reads that fully contains the whole adapter sequence. Also, AdapterRemoval can find more whole adapter sequences than zgrep does because AdapterRemoval is able to consider errors in sequencing even if one error occurs when sequencing the adapter, whereas zgrep is unable to see through such error and consider the read as having no match with the adapter.

1bi) The reason trimming the homopolymers is useful for our reads is because the homopolymers are not present in the reference genome. So, trimming the homopolymers (G, T from the left side/ 5' and A, C from the right side/3') makes aligning to the reference genome possible.

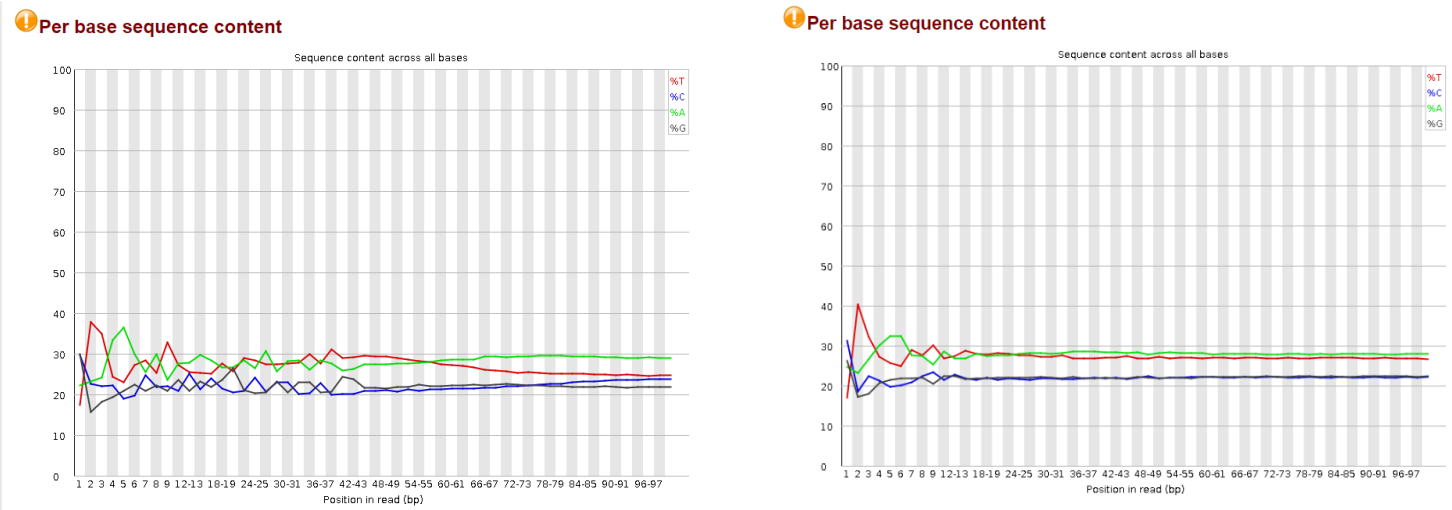
1bii)

```
[dkhor@nova dkhor]$ grep -c "Sequence removal: right side" nod1_fb/Nod1_1_S61.log
944249
[dkhor@nova dkhor]$ grep -c "Sequence removal: left side" nod1_fb/Nod1_1_S61.log
4673620
```

It does not look purely random whether a read has a match on the 5' vs the 3' end. This is because the the count for a read to have a match on the 5' end is 4673620, whereas the count for a read to have a match on the 3' end is 944249. The count for a read to have a match on the 5' end is approximately 500% of the the count for a read to have a match on the 3' end, which explains the nonrandomess. I would expect it to be nonrandom because there will generally be a match in the one read but not the other if the fragment length is greater than 101 base pair, which is why there is more matches on the left side (5') than the right side (3') of the reads in this case. This is because if the cDNA fragment length is

greater than 101 base pairs, either read 1 or read 2 will have a match at the start of its 5' end, but the other read will not be long enough to sequence the 5' end of the strand of cDNA, where the RT primer or TSO resides.

1c)



On the left is the fastqc report on the per base sequence content of the original data read file of Nod1_1_S61_L001_R1_001.fastq.gz, whereas on the right is the fastqc report on the per base sequence content of the data of read file of Nod1_1_S61_L001_R1_001.fastq.gz after it has been processed by Flexbar. As we can see, technical sequences exist in the original read, whose position is characterized by the portion of position in read that visually has zig-zag lines. These zig-zagged lines represent a fluctuation in the change of per base sequence content, meaning at each position, one particular base pair is more enriched than the others, signifying the presence of a fixed sequence, which in this case is the technical sequence. This is expected because reads on fragments that has length less than 101 base pairs will result in the technical sequences in the cDNA fragment being sequenced. On the other hand, after the reads are processed by a trimming software, we expect the per base sequence content graph of the reads to not have these zig zag lines but instead almost constant flat lines. However, that is not the case in the reads that are processed by Flexbar. As we can see, there are still zig zag lines present from the first position in the processed read to around the 13th position in the processed read. Although the length of the technical sequence has greatly reduced as compared to the original reads, which is presented by the decrease of number of continuous positions that the zig zag lines occupy, there are still indications that part of the technical sequence still exists even after the trimming process by Flexbar. Therefore, I am not convinced that the technical sequences have been removed.

2a)

```
[dkhor@nova dkhor]$ ls -l nod1_starindex
total 16746749
-rw-r--r--. 1 dkhor domain users       12389 Nov 29 22:51 chrLength.txt
-rw-r--r--. 1 dkhor domain users       41156 Nov 29 22:51 chrNameLength.txt
-rw-r--r--. 1 dkhor domain users       28767 Nov 29 22:51 chrName.txt
-rw-r--r--. 1 dkhor domain users       21137 Nov 29 22:51 chrStart.txt
-rw-r--r--. 1 dkhor domain users    27206078 Nov 29 23:00 exonGeTrInfo.tab
-rw-r--r--. 1 dkhor domain users    12221182 Nov 29 23:00 exonInfo.tab
-rw-r--r--. 1 dkhor domain users      494220 Nov 29 23:00 geneInfo.tab
-rw-r--r--. 1 dkhor domain users  2100343767 Nov 29 23:02 Genome
-rw-r--r--. 1 dkhor domain users         643 Nov 29 23:02 genomeParameters.txt
-rw-r--r--. 1 dkhor domain users 14355092593 Nov 29 23:02 SA
-rw-r--r--. 1 dkhor domain users  1565873619 Nov 29 23:02 SAindex
-rw-r--r--. 1 dkhor domain users     9393922 Nov 29 23:00 sjdbInfo.txt
-rw-r--r--. 1 dkhor domain users    10240162 Nov 29 23:00 sjdbList.fromGTF.out.tab
-rw-r--r--. 1 dkhor domain users    10239792 Nov 29 23:00 sjdbList.out.tab
-rw-r--r--. 1 dkhor domain users     4278244 Nov 29 23:00 transcriptInfo.tab
```

```
[dkhor@nova GRCz11]$ ls -l
total 1315825
-rw-rw----. 1 kdorman domain users 1700419557 Nov 22 23:57 GCF_000002035.6_GRCz11_genomic.fna
-rw-rw----. 1 kdorman domain users  590023251 Nov 22 23:58 genomic.gtf
-rw-rw----. 1 kdorman domain users  195344836 Nov 23 08:44 rna.fna
```

File size of suffix array (SA): 14355092593

File size of original genome: 1700419557

Expansion ratio between suffix array (SA) and original genome (SA / original genome)

→ 14355092593 / 1700419557 = 8.442

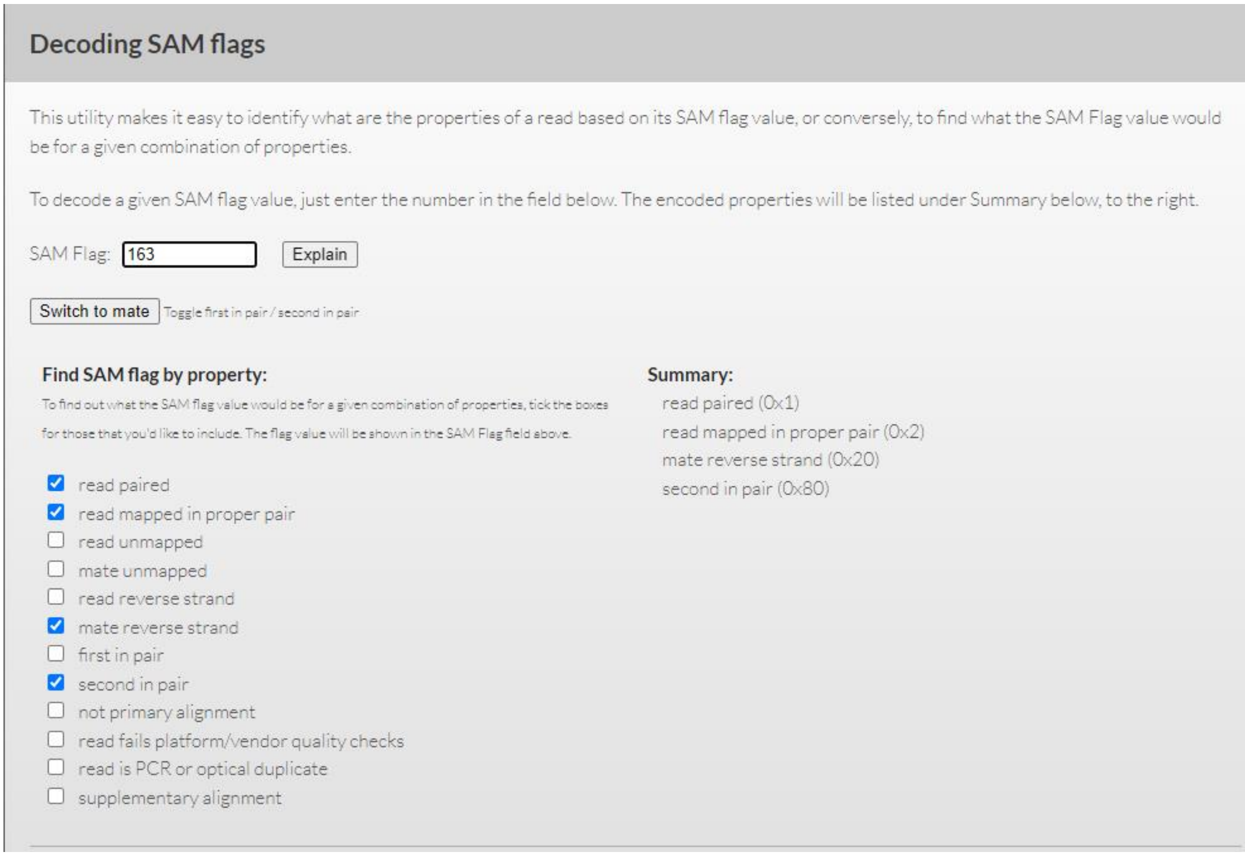
2bi)

Evidence of duplicated reads:

```
A00953:518:HL2NYDSX3:1:1160:2917:31375  163     NC_007112.7     4587    255     99M2S   =       4653    1268    TCTGCTTTATTTGGG
,F,F,,::FFF,F:FFFF:FFF,:FFF:FFFFF,F:FF::FFFFFF:F,F,,F,,:,F,,,:FF:FF,FF:,F,FFFF,F,,F:FF, NH:i:1  HI:i:1  AS:i:170        nM:i:8
A00953:518:HL2NYDSX3:1:1160:3152:31876  163     NC_007112.7     4587    255     99M2S   =       4653    1268    TCTGCTTTAGTTGGG
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFF NH:i:1  HI:i:1  AS:i:178        nM:i:4
A00953:518:HL2NYDSX3:1:1317:11550:32925 163     NC_007112.7     4587    255     99M2S   =       4653    1268    TCTGCTTTAGTTGGG
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF NH:i:1  HI:i:1  AS:i:178        nM:i:4
```

In this case, there is a total of 3 duplicates for the fragment I selected. These are duplicated reads because the position they are aligned to the reference genome are all the same (column 4: 4587). Also, the position that their paired reads are aligned to are also the same (column 8: 4653). In addition, this is

the result of decoding their SAM flags (163):



Here, we can see that these reads are indeed paired and mapped in proper pair. Therefore, I conclude that these reads, and their paired reads come from the same underlying cDNA fragment and duplication of the fragment indeed happened.

2bii)

```
[dkhor@nova dkhor]$ samtools view nod1_star/Nod1_1_S61Aligned.sortedByCoord.out.bam | grep -n A00953:518:HL2NYDSX3:1:2172:24867:23328
17:A00953:518:HL2NYDSX3:1:2172:24867:23328      99      NC_007112.7     2621    1       101M    =       2690    170     GGACCCAGGGAGGCTGAATACTTACCACTCCCCCAA
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF,FFFFFFFFFFFFFFFFFFFFFF NH:i:3  HI:i:1  AS:i:190        nM:i:5
20:A00953:518:HL2NYDSX3:1:2172:24867:23328      147     NC_007112.7     2690    1       101M    =       2621    -170    GTCACCCGAGAAGAGTCTGAAGAGGTTTCTCTATGG
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF NH:i:3  HI:i:1  AS:i:190        nM:i:5
1434382:A00953:518:HL2NYDSX3:1:2172:24867:23328 355     NC_007113.7     7235    1       101M    =       7306    170     GGACCCAGGGAGGCTGAATACTTACCACTCCCCCAA
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF,FFFFFFFFFFFFFFFFFFFFFF NH:i:3  HI:i:2  AS:i:190        nM:i:4
1434383:A00953:518:HL2NYDSX3:1:2172:24867:23328 403     NC_007113.7     7306    1       2S99M   =       7235    -170    GTCACCCGAGAAGAGTCTGAAGAGGTTTCTCTATGG
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF NH:i:3  HI:i:2  AS:i:190        nM:i:4
19478982:A00953:518:HL2NYDSX3:1:2172:24867:23328        355     NC_007124.7     120233  1       101M    =       120302  170     GGACCCAGGGAGGCTGAATACTTACCAC
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF,FFFFFFFFFFFFFFFFFFF       NH:i:3  HI:i:3  AS:i:190        nM:i:5
19478986:A00953:518:HL2NYDSX3:1:2172:24867:23328        403     NC_007124.7     120302  1       101M    =       120233  -170    GTCACCCGAGAAGAGTCTGAAGAGGTTT
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF       NH:i:3  HI:i:3  AS:i:190        nM:i:5
```

This paired read is mapped a total of 3 times. There is no one mapping that is more supported than another because all of them have the same value for mapping quality, which is 1 as indicated by column 5 (MAPQ).

3a)

This is the log summary of nod1_dedup/Nod1_1_S61.sortedByCoord.dedup.bam.

```
--------------------------------------------------------------------------------
SUMMARY STATISTICS OF THE READS
Total number of reads: 46606946
Total number of paired-end reads: 46606946
Total number of properly paired reads: 45714700
Total number of unmapped reads: 0
Total number of reverse strand mapped reads: 23246243
Total number of QC-failed reads: 0
Total number of secondary reads: 8951346
Size of singleKeyMap (must be zero): 0
Size of pairedKeyMap (must be zero): 0
Total number of missing mates: 0
Total number of reads excluded from duplicate checking: 8951346
--------------------------------------------------------------------------------
Sorting the indices of 8987559 duplicated records

Writing nod1_dedup/Nod1_1_S61.sortedByCoord.dedup.bam
Successfully removed 571405 unpaired and 4208077 paired duplicate reads
```

Percentage of reads removed because of duplication:

[(571405 + 4208077) / 46606946] * 100% = 10.3 %

3b) Let's say we are trying to test whether there is a difference in expression of gene1 in Nod1 samples that are treated with morpholino vs the control samples that are not treated with morpholino. Suppose we have a sample of counts of read pairs that aligned to gene1 at the reference genome for Nod1 samples and control samples:

$$X_1, X_2, \ldots, X_n \text{ and } Y_1, Y_2, \ldots, Y_m$$

where X_i is the count of read pairs that aligned to gene1 at the reference genome for the i^{th} Nod1 sample for i = 1,…,n; Y_j is the count of read pairs that aligned to gene1 at the reference genome for the j^{th} control sample for j = 1,…,m.

We can test whether the two samples have a difference in expression level by testing whether there is a difference in distribution between the two samples using Wilcoxon rank sum test:

$$H_0: F_X = F_Y$$

$$H_a: F_X \neq F_Y$$

By retaining PCR duplicates, it disrupts our ability to detect differential expression by increasing the false positive rate by increasing the false positive ratio, which is the probability of falsely rejecting the null hypothesis of the test. This happens because PCR duplicates increase the count of read pairs that are aligned to gene1 for each sample, which interferes with the p-value, which in turn increases our chance of coming up with the wrong conclusion about the difference in expression of the gene.

4a) I chose 'intersection-strict' for --mode, and 'all' for--nonunique. I choose "intersection-strict" and "all" because I want all the reads to be assigned to each of the corresponding gene only when the read sequence is aligned to the corresponding gene sequence such that the gene sequence contains the read sequence wholly.

4b)

```
[dkhor@nova dkhor]$ grep nod1 nod1_htseq1.txt
nod1    70      7       35      107     13      42
```

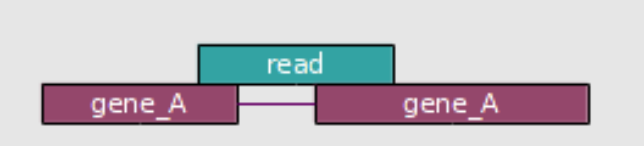
```
> wilcox.test(c(70,7,35), c(107,13,42))

        Wilcoxon rank sum exact test

data:  c(70, 7, 35) and c(107, 13, 42)
W = 3, p-value = 0.7
alternative hypothesis: true location shift is not equal to 0
```

Since the p-value is greater than 0.05 significance level, we have weak evidence against the null and thus fail to reject the null hypothesis and conclude that there is no evidence for reduced expression under treatment.

4c) The result I obtained in part 4b should not be my conclusion because the sample size is too small for the conclusion to be convincing.

4di) We should expect the number of nod1 mRNA transcript to be reduced because morpholino messes up with the splicing of nod1 pre-mRNA.

4dii) My choices for --mode as 'intersection-strict' will likely under measure nod1 expression. For example, with morpholinos treatment, resulting transcripts can have introns or only part of the exons as compared to a normal transcript that has the full exon sequences. As an example:



Some reads might contain introns due to the morpholino treatment, but "intersection-strict" does not count this read as part of the gene count for the case as shown in the figure, which makes it undercount the gene count, and thus under measures nod1 expression.