ANGSD 2020: Final Project

Effects of Smoking on Gene Expression in
South Korean Patients Patients with Lung Adenocarcinoma

Jess White

4/14/2020

Contents

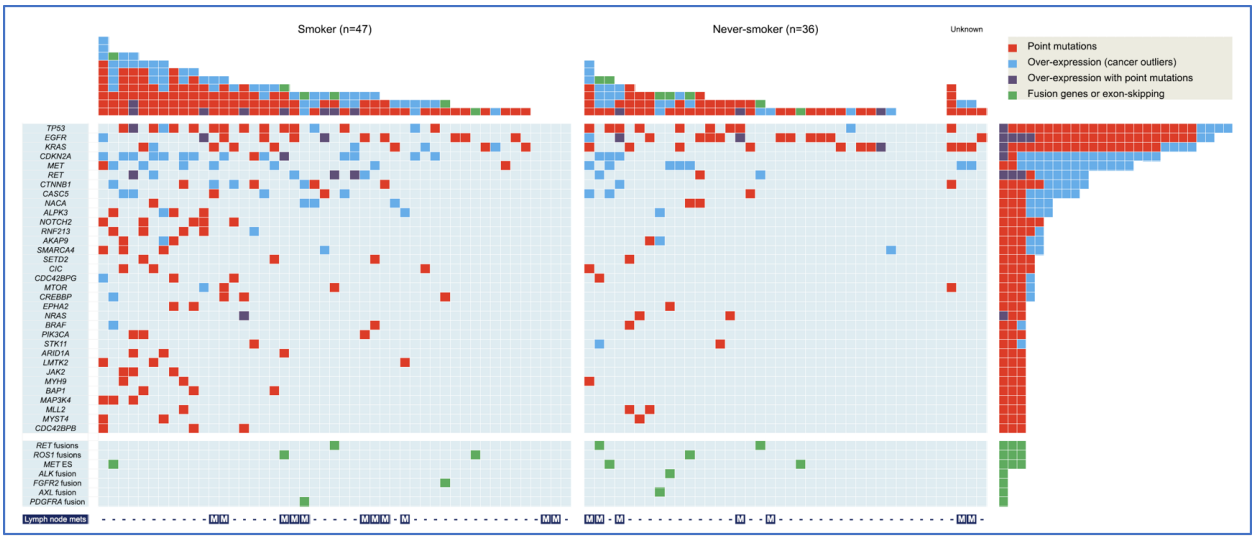
Introduction

Smoking is associated with a different mutational background in lung adenocarcinomas. For example, KRAS mutations are found in ~30% of lung adenocarcinomas but only in ~15% of lung adenocarcinomas from never-smoker lung cancer patients. Additionally, KRAS mutations associated with smoking (G12C, G12V, G12A) are more frequently transversion mutations (G>T, G>C) as opposed to transition mutations (G>A), which occur more often in never-smoker patients (Ricciuti et al., *Expert Rev Respir Med*, 2016).

Additionally, lung adenocarcinomas of smokers are associated with a higher mutational burden. In the data set used for this analysis, the following was observed:

On average, smokers had significantly more amino acid-altering single nucleotide and short-indel mutations (65.0 and 20.6 mutations per cancer tissue of smokers [n = 40] and never-smokers [n = 33], respectively; P-value = 0.0011).

Given these observations, I hypothesized that the expression profiles of lung adenocarcinomas from smokers would be different than those of non-smokers. Similarly, I postulated that the expression profiles would be different between tumor and paired-normal biopsies.



Source: Seo et al., *Genome Research*, 2012.

Figure 1: The transcriptional landscape and mutational profile of 87 lung adenocarcinomas

Methods

Summary of data

I selected data produced from 200 fresh surgical specimens of primary lung adenocarcinoma from patients who underwent major lung resection in South Korea at Seoul National University and The Catholic University of Korea. The data are available here in the ENA. A summary of the findings was originally published in Seo et al., *Genome Research*, 2012..

Data from a total of 167 samples were available from 87 individuals. The data set was broken into two sets, one prefixed with "S" containing 51 samples, including some sequenced using Sanger sequencing, and another prefixed with "C" containing 36 samples that were only sequenced using next-generation sequencing methods. I selected the "C" data for additional analysis.

Summary statistics for the cohort of 36 "C" patients is shown below. A total of 63 samples were collected from these 36 patients, including 27 paired-normal biopsies and 36 tumor biopsies.

```
##       age           gender             smoking       stage
##  Min.   :38.00   female:15   current smoker: 7   1A     :16
##  1st Qu.:54.00   male  :21   never smoker  :14   1B     : 7
##  Median :61.00               smoker        :11   2B     : 3
##  Mean   :60.47               NA's          : 4   3A     : 3
##  3rd Qu.:66.00                                   2A     : 2
##  Max.   :85.00                                   (Other): 3
##                                                  NA's   : 2
```

RNAs were extracted from tissue using RNAiso Plus (Takara Bio Inc.), followed by purification using RNeasy MinElute (Qiagen Inc.). RNAs were assessed for quality and was quantified using an RNA 6000 Nano LabChip on a 2100 Bioanalyzer (Agilent Inc.). The libraries were prepared per protocol described in Ju et al., *Nature Genetics*, 2011. Illumina HiSeq 2000 was used as the sequencing platform.

Alignment

All relevant scripts are provided via link to my GitHub repository for this project and can be accessed via the Scripts table in the Appendix.

After bulk downloading the relevant samples via a script, I wrote a script to run STAR aligner with paired end reads in separate `fastq` files. I chose to use STAR instead of BWA given that this is RNA-seq data that will likely contain splice sites. Given that STAR uses soft-clipping, I did not perform any adapter trimming. I altered some of the default parameters, including:

- `--outFilterMultimapNmax 20` to better align with ENCODE RNA-seq guidelines
- `--alignSJoverhangMin 8` to better align with ENCODE RNA-seq guidelines
- `-- alignIntronMin 10` as Luce mentioned in class, this default is considered not well chosen, so I reduced the minimum intron length to be more permissive
- `--twopassMode Basic` to find novel splice sites and run a second time to potentialy detect additional
- `--outReadsUnmapped Fastx` to collect the unmapped reads to check against BLAST

Quality control

FastQC

I created separate outputs for the paired-end read 1 and paired-end read 2 files, originally given limitations of the size constraints of the version of `FastQC` I was using. Doing so revealed that the quality of the paired-end read 2 files was much worse than that of the paired-end read 1 files. This was true for both the cancer biopsies, which generally demonstrated worse quality scores, and paired-normal tissue samples, suggesting a systemic, technical issue with the quality of the paired-end read 2 files. This is an older data set, which was published in 2012 but presumably the samples were collected and analyzed even earlier than that (no dates provided in publication). The relatively poor quality suggests perhaps the reagents or the sequencer itself (e.g., flow cells, hybridization procedure, etc.) were not as robust as they are today.

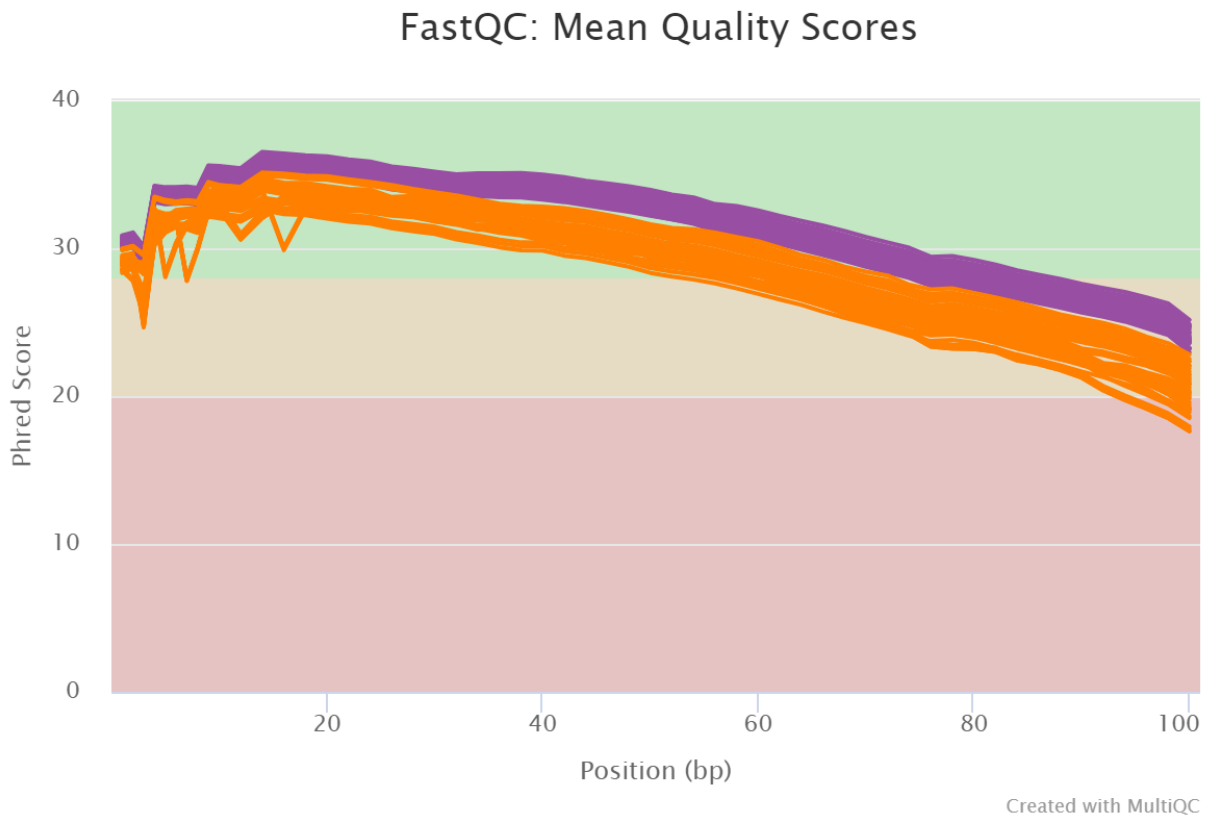


Figure 2: FastQC per base sequence quality plot for paired-end read 1 files. Orange samples are tumor biopsies and purple samples are paired-normal biopsies

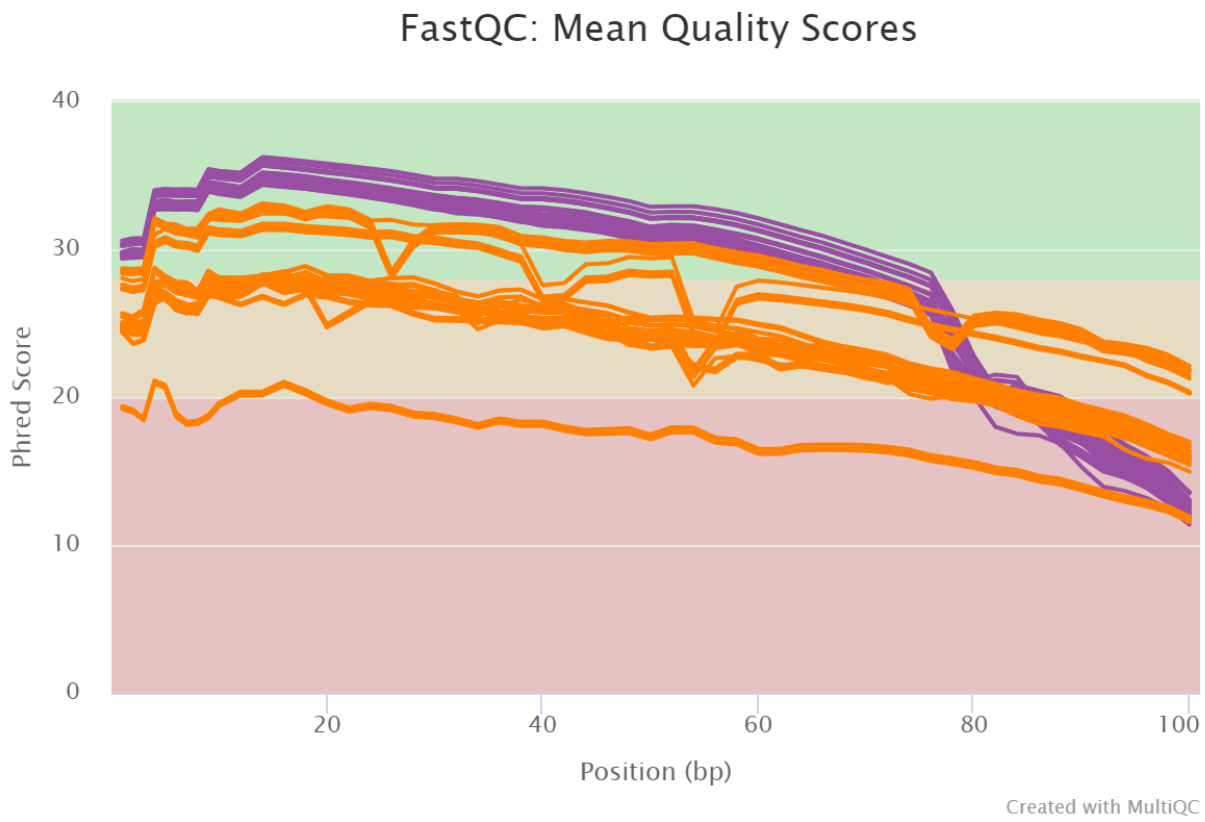


Figure 3: FastQC per base sequence quality plot for paired-end read 2 files. Orange samples are tumor biopsies and purple samples are paired-normal biopsies

Despite these observations, I did not choose to exclude the paired-end 2 read files. It is important to be aware of these limitations when assessing the quality of the observations in the downstream analyses.

STAR QC

During the alignment, I noticed that a lower percentage of reads were mapping to the genome for the tumor biopsies than for the paired normal biopsies. Given that the total number of reads was slightly higher for the tumor biopsies, the total number of reads mapped were comparable for the tumor and paired-normal biopsies.

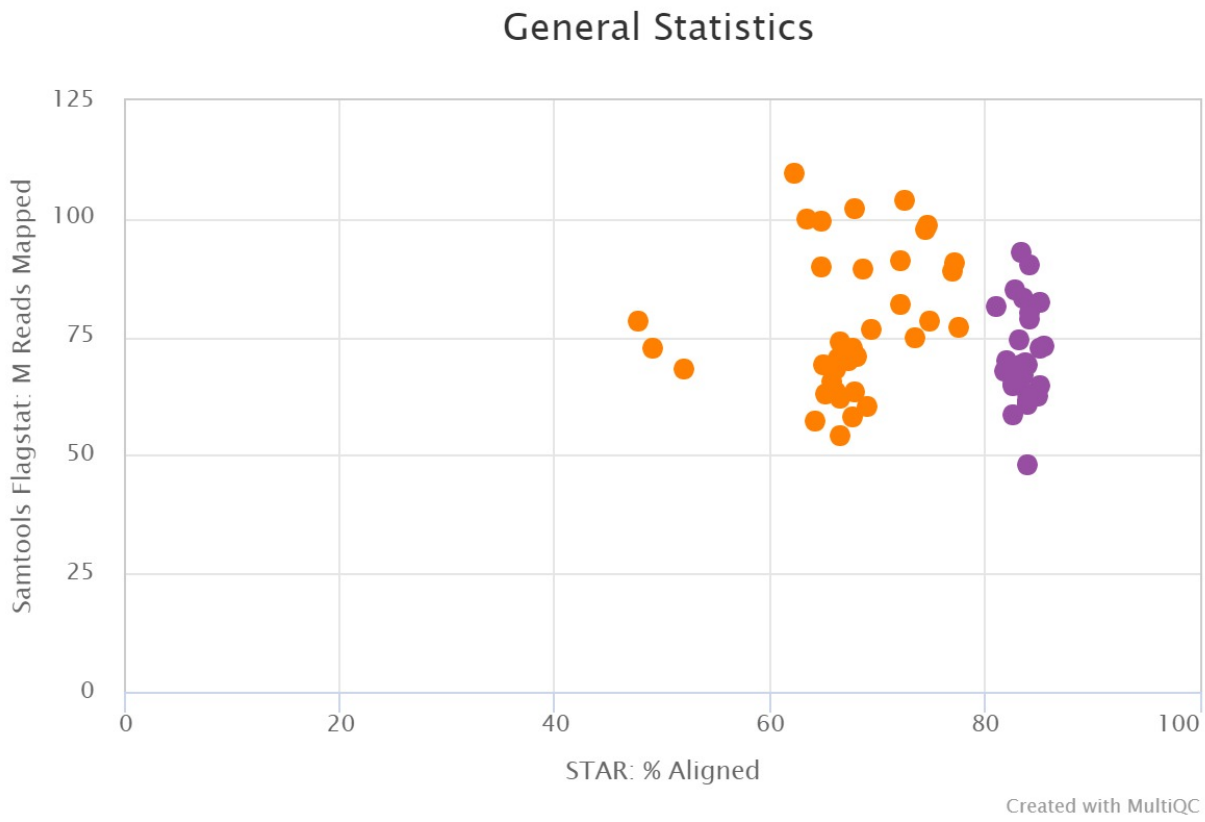


Figure 4: Millions of reads mapped vs. % aligned

Once I saw the cancer reads were mapping less frequently using STAR, I added the unmapped reads as an output to the `star_updated.sh` script and re-ran for three cancer samples (ERR164583, ERR164584, and ERR164585) and converted the `fastq` ouput to a `fasta` output compatible with BLAST. I attemped to input the entire resulting `fasta` file for each set of paired-end reads for ERR164585, but my connection timed out. Instead, I extracted the last 10 unmapped reads of from each paired-end `fasta` file. The result was that while no significant similarity was found in the paired-end 2 file, 4 of the 10 unmapped reads in the paired-end 1 file produced significant alignments.

Obviously this is only a small subset of the overally unmapped reads, but it suggests that the quality of reads in the paired-end 2 file might not be as robust, as was observed above, and that some of the reads that failed to map using `STAR` are attributable to increased mutational burden. In some cases, particularly for read 5, this mutational burden meant that the read mapped with similarly high identity to multiple locations in the genome. This suggests in the future it may be productive to run `STAR` with more permissive mismatch criteria and a variant analysis would be particularly helpful in identifying the most likely variants found in this data set.

```
Query #4: ERR164585.41770307 0:N:  00 Query ID: lcl|Query_32913 Length: 101

Sequences producing significant alignments:
                                                                  Max    Total Query   E    Per.
Description                                                       Score  Score cover Value Ident
Homo sapiens protein disulfide isomerase family A member 4...     154    154   100%  2e-36 94.06
Homo sapiens protein disulfide isomerase family A member 4...     154    154   100%  2e-36 94.06
Homo sapiens protein disulfide isomerase family A member 4...     154    154   100%  2e-36 94.06
Homo sapiens protein disulfide isomerase family A member 4...     154    154   100%  2e-36 94.06
Homo sapiens protein disulfide isomerase family A member 4...     154    154   100%  2e-36 94.06
Homo sapiens protein disulfide isomerase family A member 4...     154    154   100%  2e-36 94.06
Homo sapiens protein disulfide isomerase family A member 4...     154    154   100%  2e-36 94.06
Homo sapiens protein disulfide isomerase family A member 4...     154    154   100%  2e-36 94.06
```

Figure 5: Abbreviated list of significant alignments for paired-end file 1 ERR164585 read #4

```
Alignments:
>Homo sapiens protein disulfide isomerase family A member 4 (PDIA4), transcript variant 5, non-coding RNA
Sequence ID: NR_163906.1 Length: 2597
Range 1: 1780 to 1880

Score:154 bits(83), Expect:2e-36,
Identities:95/101(94%),  Gaps:0/101(0%), Strand: Plus/Plus

Query  1     AGAGATCTGGAGCATTTGAGCAAGTTTATAGAAGAACATGCCACAATACTGAGCAGGACT  60
             |||||||||||||||||||||||||||||||||||||||||||||| ||||||||||||
Sbjct  1780  AGAGATCTGGAGCATTTGAGCAAGTTTATAGAAGAACATGCCACAAAACTGAGCAGGACC  1839

Query  61    AAGGAAGTGCTTGTAAGGCCTGAGGTCTGCTGAAGGTGGGA  101
             ||||||| ||||  |||||||||||||||| ||||||||||
Sbjct  1840  AAGGAAGAGCTTTGAAGGCCTGAGGTCTGCGGAAGGTGGGA  1880
```

Figure 6: Suggested alignment of transcript with highest percent identity for paired-end file 1 ERR164585 read #4

RSeQC

Despite some of the differences in the `FastQC` metrics and the percentage alignment via `STAR`, the read distribution per `RSeQC` was quite similar with a slight majority of reads mapping to a coding exon region.

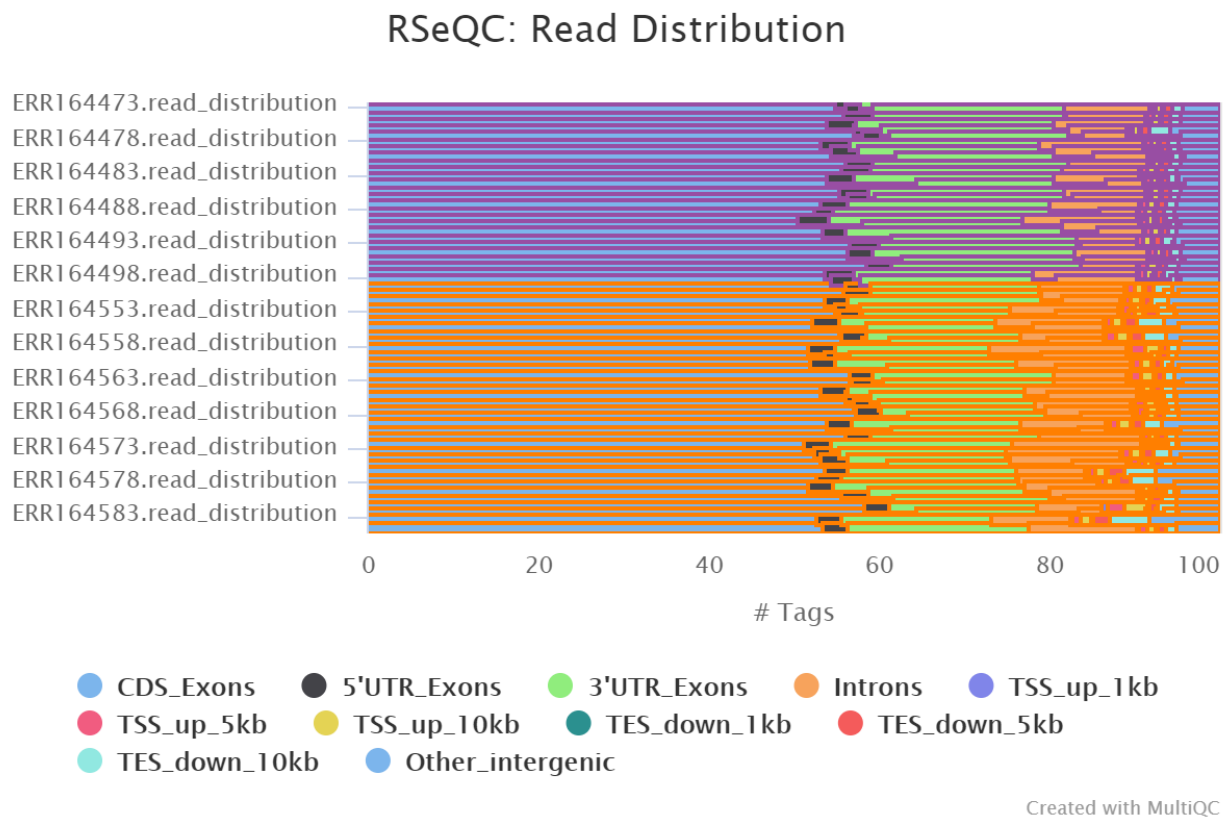


Figure 7: RSeqQC read distribution of mapped reads

I was unable to run the `geneBody_coverage.py` script for the `BAM` files given the time duration and memory requirements, but I did write a script to perform and queue this task. It would have been helpful to assess if there was degradation in the sample that could have caused the unfavorable `FastQC` findings.

Downstream analyses

```
library(magrittr)

# import read counts file and remove extraneous info in colnames
readcounts <- read.table('C:/Users/jessb/OneDrive/MS-CB/ANGSD/project/counts/project_20.04.09.txt',
                         sep = '\t',header = TRUE)
names(readcounts) = gsub(pattern = "alignment.", replacement = "", x = names(readcounts))
names(readcounts) = gsub(pattern = ".Aligned.sortedByCoord.out.bam",
                         replacement = "", x = names(readcounts))

# import column names in order
col_names <- read.table('C:/Users/jessb/OneDrive/MS-CB/ANGSD/project/files/col_names.txt',
                        header = FALSE, stringsAsFactors = FALSE)
orig_names <- names(readcounts)
```

```
rownames(readcounts) <- readcounts$Geneid
colnames(readcounts) <- c(colnames(readcounts)[1:6],
                          col_names$V1)

# remove first 6 columns and create sample conditions df
readcounts <- readcounts[ , -c(1:6)]
sample_info <- data.frame(condition = c(rep("normal", 27), rep("cancer", 36)))
rownames(sample_info) <- names(readcounts)
```

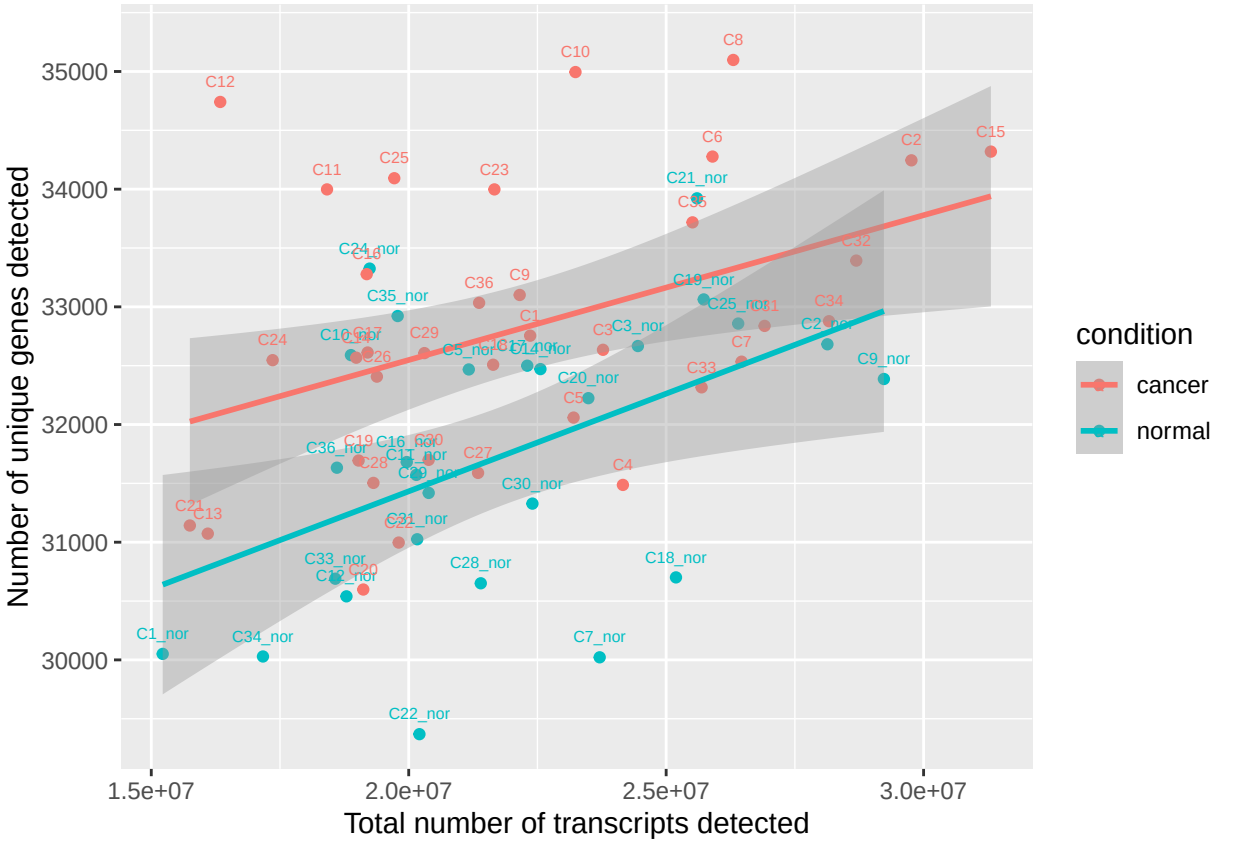
Read count assessment

Given that I observed that during alignment that a lower percentage of transcripts mapped to the genome, I wanted to see how the total number of transcripts compared to the number of unique genes detected for the cancer and paired-normal biopsies. Even though a lower percentage of reads mapped for the cancer biopsies, both cancer and normal samples contained between between 15 and 30 million transcripts. However, the cancer samples contained many more uniquely mapped genes. I ran a multiple regression of number of unique genes detected on number of total transcripts detected and the biopsy source type, and even controlling for the total number of transcripts detected, the condition had a statistically significant impact on the number of unique genes detected.

```
library(ggplot2)
library(reshape2)

col_sums <- melt(colSums(readcounts))
condition <- c(rep("normal", 27), rep("cancer", 36))
col_sums <- cbind(col_sums, condition)
for (n in seq(1, length(readcounts))){
  col_sums$genes_expressed[n] <- sum(readcounts[, n] > 0)
}
rownames(col_sums) <- gsub("LC_", "", rownames(col_sums))

ggplot(col_sums, aes(x = value, y = genes_expressed, color = condition)) +
  geom_point() +
  geom_smooth(method = lm) +
  geom_text(label=rownames(col_sums), size=2, nudge_y = 175) +
  xlab("Total number of transcripts detected") +
  ylab("Number of unique genes detected")
```




```
summary(lm(genes_expressed ~ value + factor(condition), data = col_sums))
```

```
##
## Call:
## lm(formula = genes_expressed ~ value + factor(condition), data = col_sums)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2142.6  -781.3    49.2   602.1  2730.7
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                2.975e+04  8.324e+02  35.746  < 2e-16 ***
## value                      1.381e-04  3.667e-05   3.765 0.000381 ***
## factor(condition)normal -1.033e+03  2.735e+02  -3.776 0.000368 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1073 on 60 degrees of freedom
## Multiple R-squared:  0.3323, Adjusted R-squared:   0.31
## F-statistic: 14.93 on 2 and 60 DF,  p-value: 5.464e-06
```

DESeq analysis

I performed the preliminary `DESeq` assessment using tumor vs. paired-normal biopsy as the design condition.

```
library(DESeq2)

DESeq.ds <- DESeqDataSetFromMatrix(countData = readcounts,
                                   colData = sample_info,
                                   design = ~ condition)

# drop genes with no reads
keep_genes <- rowSums(counts(DESeq.ds)) > 0
DESeq.ds <- DESeq.ds[ keep_genes, ]

#rlog normalization
DESeq.rlog <- rlog(DESeq.ds, blind = TRUE)
rlog.norm.counts <- assay(DESeq.rlog)
save.image(file = "C:/Users/jessb/OneDrive/MS-CB/ANGSD/project/RNAseqSeo.RData")
```

Clustering analysis

I created a heatmap and a dendrogram using complete linkage hierarchical clustering. In the dendrogram, four major clusters emerged. The first cluster on the left showed the highest degree of similarity. Interestingly, 89% of the 9 patients in this cluster identified as smokers. The next cluster contained 13 patients of which 69% identified as never smokers. All but one of the paired normal biopsies clustered together. Finally, there was a cluster that contained a mix of smokers, ever smokers, and NAs. Even within this less well-defined cluster, two of the sub-clusters contained either all smokers or never smokers. Interestingly, this was the cluster with the lowest degree of similarity.

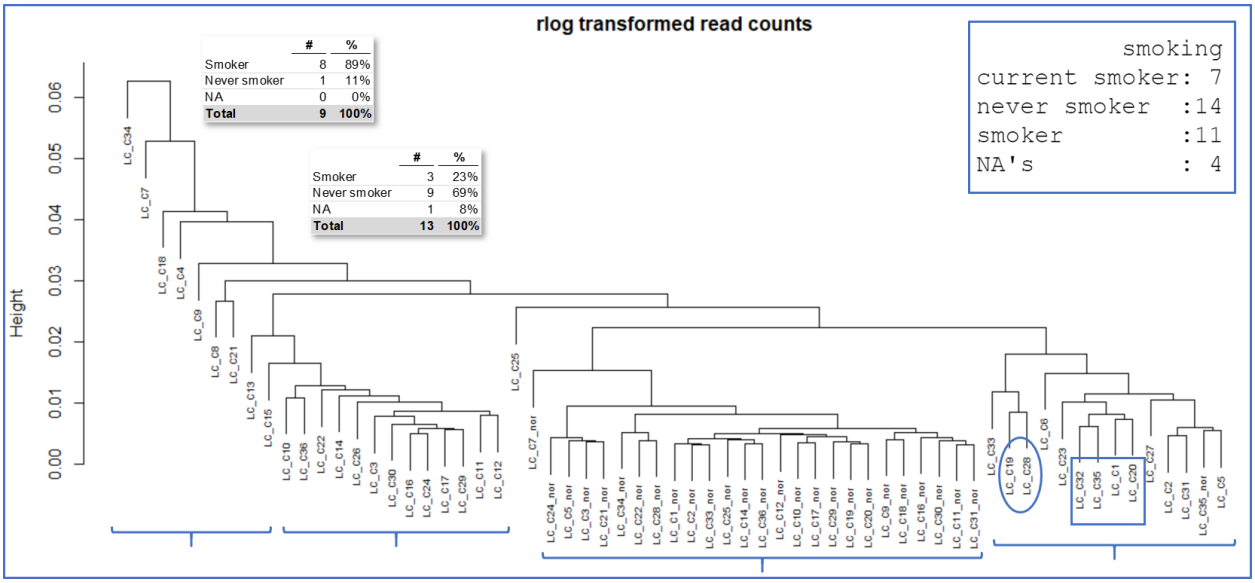
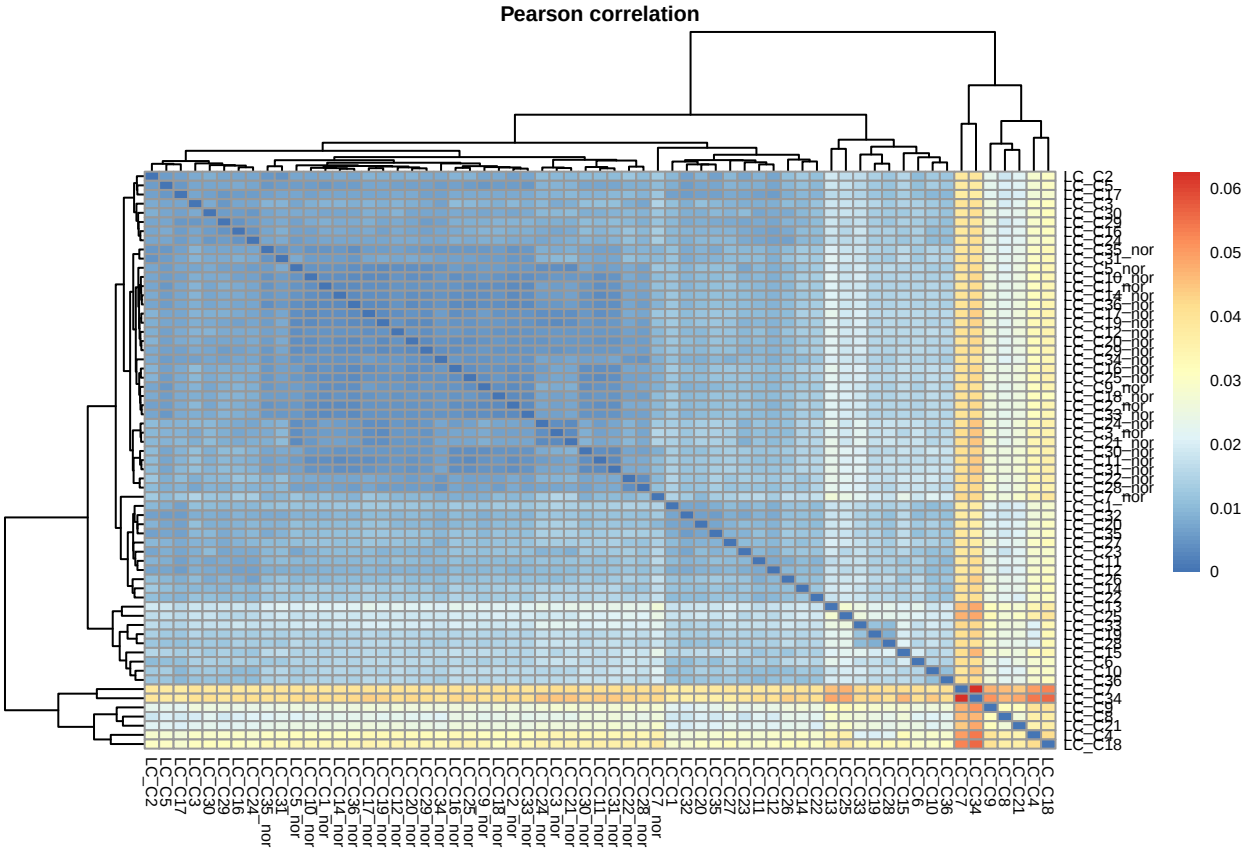


Figure 8: RSeqQC read distribution of mapped reads

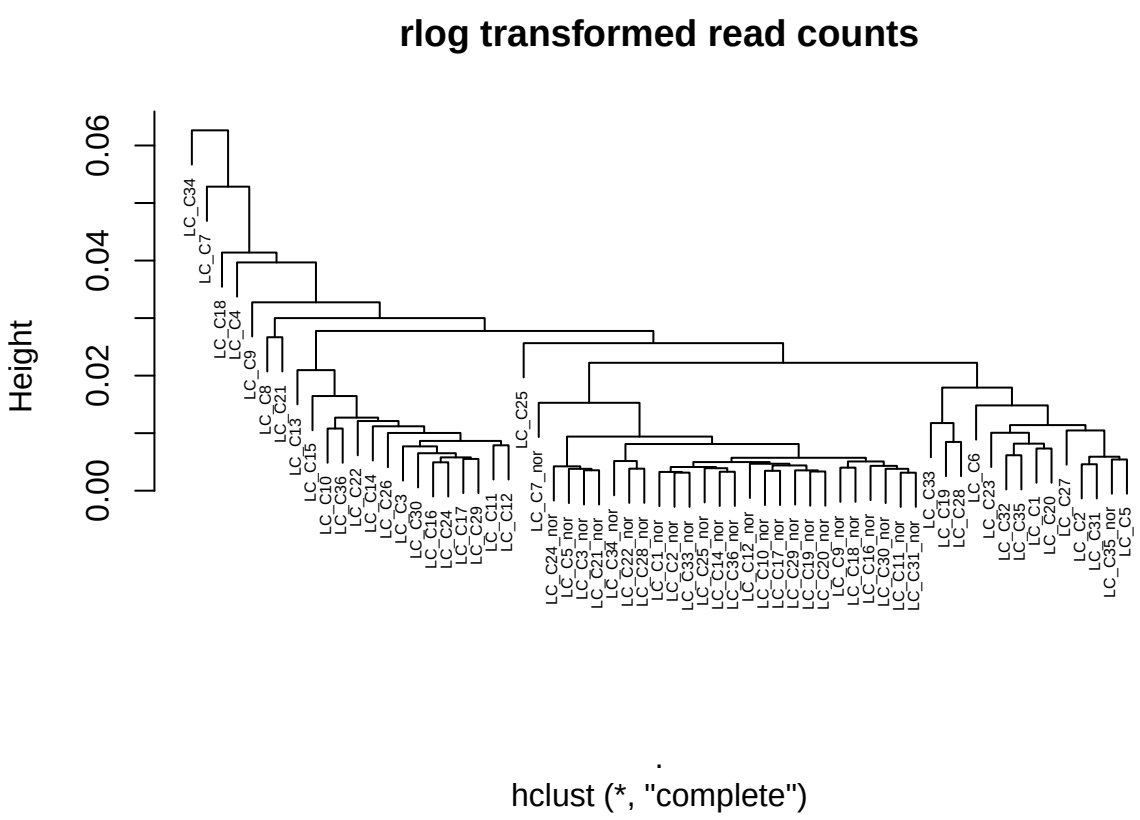
```
load("C:/Users/jessb/OneDrive/MS-CB/ANGSD/project/RNAseqSeo.RData")
library(pheatmap)
library(magrittr)

corr_coeff <- cor(rlog.norm.counts, method = "pearson")
fontsize = 10 - ncol(readcounts) / 15
as.dist(1-corr_coeff, upper = TRUE) %>%
```

```
  as.matrix %>%
  pheatmap(., main = "Pearson correlation", fontsize = fontsize)
```




```
as.dist(1 - corr_coeff) %>% hclust %>%
  plot( ., labels = colnames(rlog.norm.counts),
        main = "rlog transformed read counts", cex = 0.5)
```



Differential gene expression analysis

Using the condition of tumor or paired-normal biopsy, over 18,000 individual genes showed significant differential expression after adjusting for multiple hypothesis correction.

```
library(DESeq2)

DESeq.ds <- DESeq(DESeq.ds)
DGE.results <- results(DESeq.ds, independentFiltering = TRUE, alpha = 0.05)

head(DGE.results)
```

```
## log2 fold change (MLE): condition normal vs cancer
## Wald test p-value: condition normal vs cancer
## DataFrame with 6 rows and 6 columns
##                           baseMean      log2FoldChange             lfcSE
##                          <numeric>           <numeric>         <numeric>
## ENSG00000223972  0.615781833784067  -0.711203168865135 0.764616298101547
## ENSG00000227232   67.5924947329791 -0.0350914075895856 0.109723305208789
## ENSG00000278267   11.5484168022757  -0.598956713670823 0.202776454673201
## ENSG00000243485  0.126255590704116  -0.152484438997839  1.60592505645207
## ENSG00000237613 0.0818142505556582 -0.0990533924676239  2.17498602727382
## ENSG00000268020 0.0349879793497242   0.141389413044116  2.96117348450847
##                              stat              pvalue               padj
##                         <numeric>           <numeric>          <numeric>
## ENSG00000223972  -0.930143878218355   0.352296594930826   0.45148030030895
## ENSG00000227232  -0.319817266922567    0.74910685743167   0.811764368026091
```

```
## ENSG00000278267   -2.95377840901753 0.00313909316617966 0.00816205176803138
## ENSG00000243485 -0.0949511550275696   0.924353624193159                  NA
## ENSG00000237613 -0.0455420822136407   0.963675232921813                  NA
## ENSG00000268020  0.0477477641157474     0.9619172672286                  NA
```

```
summary(DGE.results)
```

```
##
## out of 52006 with nonzero total read count
## adjusted p-value < 0.05
## LFC > 0 (up)       : 6686, 13%
## LFC < 0 (down)     : 11387, 22%
## outliers [1]       : 0, 0%
## low counts [2]     : 16153, 31%
## (mean count < 0)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results
```

```
table(DGE.results$padj < 0.05)
```

```
##
## FALSE  TRUE
## 17801 18073
```

Discussion of differentially expressed genes

Many of the genes that emerged as signficantly differentially expressed between cancer and normal tissue at the lowest p-values were either pseudogenes or lncRNA. Study of these genetic elements is emerging, so assessment of their importance is challenging. Of the different coding regions that were differentially expressed, a number have been previously shown to be differentialy expressed in lung adenocarcinoma, including EBF1 (Shen et al., *Cancer Res*, 2019), MTBP (in the p53 pathway; Mao et al., *Medicine*, 2018), CCL19 (important for the tumor microenvironment; Cheng et al., *J Allergy Clin Immunol*, 2018), ZNF683, MIR664A (associated with smoking-induced COPD; Zhong et al., *Int J Chron Obstruct Pulmon Dis*, 2019).

```
# BiocManager::install("EnsDb.Hsapiens.v86")
library(EnsDb.Hsapiens.v86)

sig_genes <- order(DGE.results$padj)[1: length(which(DGE.results$padj < 0.05))]
sig_genes <- DataFrame(gene_index = sig_genes)
sig_genes$geneid <- rownames(readcounts)[sig_genes$gene_index]
sig_genes$padj <- sort(DGE.results$padj)[1: length(which(DGE.results$padj < 0.05))]

geneIDs1 <- ensembldb::select(EnsDb.Hsapiens.v86, keys = sig_genes$geneid,
                              keytype = "GENEID", columns = c("SYMBOL","GENEID"))

sig_genes_new <- merge(sig_genes, geneIDs1, all.x = TRUE,
                       by.x = "geneid", by.y = "GENEID")
sig_genes_new <- sig_genes_new[order(sig_genes_new$padj), ]

library(knitr)
top_25 <- head(sig_genes_new, 25)
top_25$padj <- format(top_25$padj, digits = 3)
kable(top_25)
```

geneid	gene_index	padj	SYMBOL
ENSG00000164330	18061	1.56e-59	EBF1
ENSG00000234518	2227	1.24e-56	PTGES3P1
ENSG00000175166	12546	1.46e-53	PSMD2
ENSG00000259447	44853	5.33e-51	RP11-462P6.1
ENSG00000262855	49277	2.07e-45	RP11-46I8.4
ENSG00000287199	15916	3.33e-45	NA
ENSG00000172167	29044	4.42e-45	MTBP
ENSG00000172724	30008	4.95e-45	CCL19
ENSG00000282975	11595	6.30e-44	RP11-59J16.3
ENSG00000254781	32229	9.07e-44	GVINP2
ENSG00000235232	8015	4.98e-43	MRPS18BP2
ENSG00000245729	15688	3.44e-41	RP11-480D4.1
ENSG00000287006	34667	6.55e-41	NA
ENSG00000260806	43719	7.40e-41	RP11-872J21.3
ENSG00000176083	868	1.02e-40	ZNF683
ENSG00000281696	4777	2.96e-40	MIR664A
ENSG00000183631	26428	9.01e-40	PRR32
ENSG00000260472	47686	2.76e-39	CTD-2358C21.2
ENSG00000261293	47023	3.68e-39	RP11-276H1.2
ENSG00000150995	9719	5.03e-39	ITPR1
ENSG00000274937	45747	5.06e-39	CTD-2311M21.4
ENSG00000274210	3056	2.10e-38	U1
ENSG00000257668	39093	2.87e-38	RP11-58A17.2
ENSG00000252943	29739	2.90e-38	RNU6-264P
ENSG00000270241	1127	3.79e-38	RP4-657M3.2

```
# write.csv(sig_genes_new, "sig_genes.csv")
```

Appendix

Scripts

Script	**Description**
ANGSD_webscraping.ipynb	Allows collection of per sample smoking and cancer vs. paired-normal data
fastqc.sh	Runs FastQC on a single FASTQ file
flagstat.sh	Runs Flagstat on a single BAM file
geneBody.sh	Runs RSeQC geneBody_coverage (took too long to run to incorporate into output)
get_files.sh	An sbatch script to download all FASTQ files from the SRA (doesn't check for continuation)
get_files_queue	An sbatch script to download all FASTQ files from the SRA (does check for continuation)
queue_fastqc.sh	Queues the FastQC jobs individually from ERR_list
queue_flagstat.sh	Queues the Flagstat jobs individually from ERR_list
queue_star.sh	Rakes ERR_list and creates individual sbatch commands to run STAR alignment

Script	Description
queue_geneBody.sh	Queues geneBody_coverage jobs individually from ERR_list
queue_rseqc.sh	Queues read_distribution jobs individually from ERR_list
queue_star.sh	Queues STAR jobs individually from ERR_list
read_counts.sh	Runs FeatureCounts (could not get this to work)
rseqc.sh	Runs RSeQC read_distribution
star.sh	Basic 1-pass STAR script
star_for_loop.sh	Runs STAR 1-pass as for loop
star_updated.sh	Run STAR alignment using basic 2-pass

Interactive code

Setting up GitHub repo

On github, I created an ANGSD repo without a `README` file. I already had git initialized in my local ANGSD directory, so I used the following commands to link the local directory to the github repo.

```
cd OneDrive/MS-CB/ANGSD
git add .
git commit -m "commit with full folder"
git remote add origin https://github.com/jessicaw9910/ANGSD.git
git remote -v
git push origin master
```

Create ERR and FTP lists to download data

```
wget -O experiment_list.txt 'https://www.ebi.ac.uk/ena/data/warehouse/filereport?accession=PRJEB2784&res
egrep "ERR" experiment_list.txt | cut -f 5 > ERR_list.txt
cut -f 6 experiment_list.txt | egrep "ftp" > ftp_links.txt

# create one directory per each sample and check number of folders in wd
for recs in `cat ERR_list.txt`;
  do mkdir ${recs};
done
ls -l | grep "^d" | wc -l

# looking at the experiment_list.txt file, there are 5 run_accessions without associated FASTQ files
rmdir ERR062334 ERR062335 ERR062336 ERR062337 ERR062338

# then used text editor to remove those 5 files from the ERR_list.txt
nano ERR_list.txt

# paired end reads in separate FASTQ files
cat ftp_links.txt | grep -o "^.*;" | sed 's/.$//' > ftp_1.txt
cat ftp_links.txt | egrep ";" | sed 's/^.*;//' > ftp_2.txt
```

Subsetting ERR list to test scripts

```
# ran code using following commands (sed to break into smaller subsets)
sed -n '1,3p;4q' ERR_list.txt > ERR_list_rev.txt
sbatch queue_star.sh ERR_list_rev.txt
```

Annotation file download

```
cd ..
wget ftp://ftp.ensembl.org/pub/release-99/gtf/homo_sapiens/Homo_sapiens.GRCh38.99.gtf.gz -O hg38.99.gtf
```

STAR: hg38 index creation

I created an index with the hg38 genome using the following commands.

```
wget http://hgdownload.soe.ucsc.edu/goldenPath/hg38/bigZips/hg38.fa.gz
gunzip hg38.fa.gz
wget http://hgdownload.soe.ucsc.edu/goldenPath/hg38/bigZips/genes/hg38.ensGene.gtf.gz
gunzip hg38.ensGene.gtf.gz
mkdir hg38_STARindex

STAR -- runMode genomeGenerate \
     -- runThreadN 8 \
     -- genomeDir hg38_STARindex \
     -- genomeFastaFiles hg38.fa \
     -- sjdbGTFfile hg38.ensGene.gtf
```

Unmapped reads

Once I saw the cancer reads were mapping less frequently using STAR, I added the unmapped reads as an output to the `star_updated.sh` script and re-ran for three cancer samples (ERR164583, ERR164584, and ERR164585).

```
sed -n '1,10p;11q' ERR_cancer.txt > ERR_list_rev.txt

cd $DIR/alignment/test
spack find | egrep "fastx"
spack load fastx-toolkit@0.0.14
fastq_to_fasta -n -i ERR164585.Unmapped.out.mate1 -o ERR164585.Unmapped.out.mate1.fasta
fastq_to_fasta -n -i ERR164585.Unmapped.out.mate2 -o ERR164585.Unmapped.out.mate2.fasta
```

FastQC

```
export TIME='\t%E real,\t%U user,\t%S sys,\t%K amem,\t%M mmem'

spack load fastqc

/usr/bin/time for dir in `cat ERR_cancer.txt`;
  do fastqc ./${dir}/${dir}_1.fastq.gz \
  fastqc ./${dir}/${dir}_2.fastq.gz;
done
```

```
4:37.39 real,   257.60 user,    21.49 sys,      0 amem, 203648 mmem
```

This suggested that `fastqc.sh` should request 25GB of memory.

BamQC

```
/softlib/apps/EL7/BamQC/bin/bamqc -o QC/bamqc -f hg38.99.gtf.gz -g hg38_STARindex/ alignment/*.bam
```

RSeQC: read distribution

```
wget https://sourceforge.net/projects/rseqc/files/BED/Human_Homo_sapiens/hg38_RefSeq.bed.gz/download -O
gunzip $DIR/hg38_RefSeq.bed.gz
```

```
# create symbolic link to class scratch directory - did this last time so no need to repeat
#ln -s /athena/angsd/scratch/jwh4001 /home/jwh4001/angsd
RSEQC_IMAGE="/athena/angsd/scratch/simg/rseqc-3.0.1.simg"
cd ~
spack load singularity@2.6.0
BED_FILE="/home/jwh4001/angsd/jwh4001/project/hg38_RefSeq.bed"
BAM_FILE="/home/jwh4001/angsd/jwh4001/project/alignment_2/*.bam"
OUT_FILE="/home/jwh4001/angsd/jwh4001/project/QC/project.read_distribution.txt"
singularity exec $RSEQC_IMAGE read_distribution.py -r $BED_FILE -i $BAM_FILE >> $OUT_FILE
```

Also ran RSeqQC `read_distribution.py` using a for loop.

```
# batched ~10 at a time to confirm was running properly
sed -n '1,10p;11q' ERR_cancer.txt > ERR_list_rev.txt

BED_FILE="/home/jwh4001/angsd/jwh4001/project/hg38_RefSeq.bed"
BAM_DIR="/home/jwh4001/angsd/jwh4001/project/alignment"
OUT_DIR="/home/jwh4001/angsd/jwh4001/project/QC/rseqc"

RSEQC_IMAGE="/athena/angsd/scratch/simg/rseqc-3.0.1.simg"

spack load singularity@2.6.0

for ERR in `cat ERR_list_rev.txt`; do
    singularity exec $RSEQC_IMAGE read_distribution.py -r $BED_FILE -i \
    ${BAM_DIR}/${ERR}.Aligned.sortedByCoord.out.bam >> \
    ${OUT_DIR}/${ERR}.read_distribution.txt;
done
```

Samtools: Flagstat

```
spack load samtools@1.9%gcc@6.3.0
cd ../alignment
for n in *.bam; do
  name="${n%.Aligned.sortedByCoord.out.bam}"
  samtools flagstat ${n} > ${name}.flagstat.txt;
done
mv *flagstat.txt ../QC/

# as a test to allow me to write a script allocating enough memory
/usr/bin/time samtools flagstat ../alignment/ERR164585.Aligned.sortedByCoord.out.bam > ../QC/flagstat/E

1:27.13 real,   82.03 user,     1.92 sys,       0 amem, 3376 mmem
```

This suggested that `flagstat.sh` should request 5GB of memory.

MultiQC report

```
cd $DIR/QC/

spack load -r py-multiqc

multiqc -n multiqc_report_2020.04.12.html ../alignment/*.Log.final.out flagstat/*.flagstat.txt fastqc/*

cp multiqc_report.html ~
```

```
# could not figure out how to incorporate BamQC reports into MultiQC
find . -type f -name '*.zip' -exec unzip -- '{}' -x '*.zip' \;
/BamQC_files
```

featureCounts

I used most of the `featureCounts` defaults, with the exception of `-p`, which I added, denoting that the reads are paired-end. Note that I attempted to write a script as occassionally my connection timed out when I ran the below code, but I received an error message directing me to the `featureCounts` documentation when I ran an identical script to the below (adjusting directory location where necessary).

```
mkdir read_counts

cd read_counts

var=$(date +'%y.%m.%d')

spack load subread

# create featureCounts files for the 5 files I was able to align
featureCounts -p \ #since paired-end reads
              -T 2 \
              -a hg38.99.gtf.gz \
              -o read_counts/project_${var}.txt \
              alignment/*.bam

cp read_counts/project_20.03.14.* ~
```

Downloading files from SCU and uploading to GitHub

To download the following files and upload to my repo I used the following commands.

```
cp project_*.txt* ~
pscp jwh4001@aristotle.med.cornell.edu:/home/jwh4001/multiqc_report.html OneDrive/MS-CB/ANGSD/project
pscp jwh4001@aristotle.med.cornell.edu:/home/jwh4001/project_*.txt* OneDrive/MS-CB/ANGSD/project

cd OneDrive/MS-CB/ANGSD/project
git add .
git commit -m "updated multiqc and read count files"
git push origin master
```

References

Cheng, H.W., et al. CCL19-producing fibroblastic stromal cells restrain lung carcinoma growth by promoting local antitumor T-cell responses. J Allergy Clin Immunol 142, 1257-1271.e1254 (2018).

Hou, H., Sun, D. & Zhang, X. The role of MDM2 amplification and overexpression in therapeutic resistance of malignant tumors. Cancer Cell Int 19, 216 (2019).

Ju, Y.S., et al. Extensive genomic and transcriptional diversity identified through massively parallel DNA and RNA sequencing of eighteen Korean individuals. Nat Genet 43, 745-752 (2011).

Mao, Y., et al. Hyper expression of MTBP may be an adverse signal for the survival of some malignant tumors: A data-based analysis and clinical observation. Medicine (Baltimore) 97, e12021 (2018).

Ricciuti, B., et al. Targeting the KRAS variant for treatment of non-small cell lung cancer: potential therapeutic applications. Expert Rev Respir Med 10, 53-68 (2016).

Seo, J.S., et al. The transcriptional landscape and mutational profile of lung adenocarcinoma. Genome Res 22, 2109-2119 (2012).

Shen, A., et al. EBF1-Mediated Upregulation of Ribosome Assembly Factor PNO1 Contributes to Cancer Progression by Negatively Regulating the p53 Signaling Pathway. Cancer Res 79, 2257-2270 (2019).

Zhong, S., et al. Overexpression Of hsa-miR-664a-3p Is Associated With Cigarette Smoke-Induced Chronic Obstructive Pulmonary Disease Via Targeting FHL1. Int J Chron Obstruct Pulmon Dis 14, 2319-2329 (2019).