STAT115/215 BIO/BST282 2021 HW3

{Your name}

Due date: 3/8,2021 @ 11:59pm

```
library(knitr)
```

Motifs, ChIP-seq, and Gene Expression Integration

Androgen receptor (AR) is a transcription factor frequently over-activated in prostate cancer. To study AR regulation in prostate cancer, scientists conducted AR ChIP-seq in prostate tumors and normal prostate tissues. Since the difference between individual patients could be quite big, this study actually included many more tumor and normal samples. However, for the purpose of this HW, we will only use the ChIP-seq data from 1 prostate tumor samples (tumor) and 1 normal prostate tissues (normal).

Hint: It helps to read the MACS README and Nature Protocol paper:
https://pypi.org/project/MACS2/
https://www.nature.com/articles/nprot.2012.101

Part I. ChIP-seq peak calling

Question 1

Usually we use BWA to map the reads to the genome for ChIP-seq experiment. We will give you one example ChIP-seq single-end sequenced .fastq file with only 1M reads. Run BWA on this file to Hg38 of the human genome assembly. Report the commands, logs files, and a snapshot / screenshot of the output to demonstrate your alignment procedure. What proportion of the reads are successfully mapped (to find at least one location) and what proportions are uniquely mapped (to find a single location) in the human genome in this test sample? We will save you some time and directly give you the BWA mapped BAM files for the full samples.

Hint:
1). Target sample fastq file is stored as **/n/stat115/2021/HW3/tumor_1M.fastq** on Cannon.
2). The index file is stored as **/n/stat115/2021/HW3/bwa_hg38_index/hg38.fasta** on Cannon.
Throughout this HW, all the coordinates are given in Hg38 version of the human genome assembly.

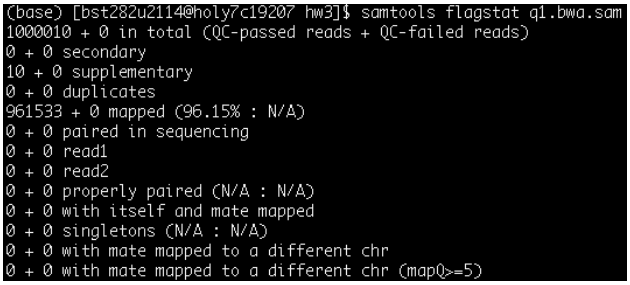
```
srun --nodes=1 --ntasks-per-node=1 --mem=64G --cpus-per-task=8 --time=02:00:00 --pty bash -i

module load bwa/0.7.15-fasrc02

# run bwa and store output on tumor_1M.fastq and store output in q1.bwa.sam
bwa mem /n/stat115/2021/HW3/bwa_hg38_index/hg38.fasta /n/stat115/2021/HW3/tumor_1M.fastq > q1.bwa.sam

# samtools for summary stats
```

```
module load samtools/1.5-fasrc02

# number of total reads and successfully mapped reads
samtools flagstat q1.bwa.sam

# filter uniquely mapped reads and save to q1.unique.bam
samtools view -bq 1 q1.bwa.sam > q1.unique.bam

# number of unique total reads and successfully mapped reads
samtools flagstat q1.unique.bam

#Answer
```

Commands

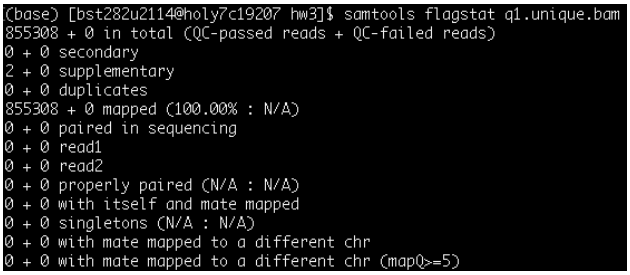
Output This is the output from samtools flagstat before I filtered to uniquely mapped reads.

```
knitr::include_graphics("q1.flagstat.bwa.png")
```



This is the output from samtools flagstat after I filtered to uniquely mapped reads.

```
knitr::include_graphics("q1.flagstat.unique.png")
```



Answer There are 1,000,010 reads in total, 961,533 of which (96.15%) were mapped. When I filtered this down into unique reads, there were 855,308 (85.5%) uniquely mapped reads.

Question 2 For Graduate Students Only

In ChIP-seq experiments, when sequencing library preparation involves a PCR amplification step, it is common to observe multiple reads where identical nucleotide sequences are disproportionally represented in the final results. This is especially a problem in tissue ChIP-seq experiments (as compared to cell lines) when input cell numbers are low. Duplicated read % is sometimes a good way to QC your ChIP-seq sample, as high duplicated reads indicate PCR over amplification of the ChIP DNA. Removing these duplicated reads can improve the peak calling accuracy. Thus, it may be necessary to perform a duplicate read removal step, which flags identical reads and subsequently removes them from the dataset. Run this on your test

sample (1M reads) (macs2 filterdup function). What % of reads are redundant? Note: when doing peak calling, MACS filters duplicated reads by default.

Hint: The test samples are stored as **/n/stat115/2021/HW3/tumor.bam** and **/n/stat115/2021/HW3/normal.bam** on Cannon.

```
module load centos6/0.0.1-fasrc01
module load macs2/2.1.2_dev-fasrc01

macs2 filterdup -i q1.unique.bam -g hs --keep-dup 1 -o q2.sample.bed

macs2 filterdup -i /n/stat115/2021/HW3/tumor.bam -g hs --keep-dup 1 -o q2.tumor.bed

macs2 filterdup -i /n/stat115/2021/HW3/normal.bam -g hs --keep-dup 1 -o q2.normal.bed
```

Running the macs2 `filterdup` function on the sample data `bed` file from question 1, the redundant rate of alignment was 0.01, so 1% of the reads were redundant.

Running the macs2 `filterdup` function on the full tumor and normal datasets, **17% of the tumor reads and 12% of normal reads were redundant**.

Question 3

For many ChIP-seq experiments, usually chromatin input without enriching for the factor of interest is generated as control. However, in this experiment, we only have ChIP (of both tumor and normal) and no control samples. Without control, MACS2 will use the non-peak read signals around the peaks to infer the chromatin background and estimate the ChIP enrichment over background. In ChIP-seq, + strand reads and − strand reads are distributed to the left and right of the binding site, respectively, and the distance between the + strand reads and − strand reads can be used to estimate the fragment length from sonication (note: with PE seq, insert size could be directly estimated). Use MACS2 to call peaks from tumor1 and normal1 separately. How many peaks do you get from each condition with FDR < 0.05 and fold change > 5? What is the estimated fragment size in each?

```
# NORMAL
macs2 callpeak -t q2.normal.bed -f AUTO -g hs -q 0.05 --fe-cutoff 5 --outdir q3.out -n normal
# standard out shows "predicted fragment length is 153 bps"

# see peak number of the last peak, shows 11780
tail -n 1 q3.out/normal_peaks.xls

# TUMOR
macs2 callpeak -t q2.tumor.bed -f AUTO -g hs -q 0.05 --fe-cutoff 5 --outdir q3.out -n tumor
# standard out shows "predicted fragment length is 159 bps"

# see peak number of the last peak, shows 28477
tail -n 1 q3.out/tumor_peaks.xls
```

Running the macs2 `callpeak` function resulted in **28,477 peaks for the tumor condition** and **11,780 peaks for the normal condition**. The **average fragment length is 159 base pairs for normal peaks and 153 base pairs for normal**.

Question 4

Now we want to see whether AR has differential binding sites between prostate tumors and normal prostates. MACS2 does have a function to call differential peaks between conditions, but requires both conditions to have input control. Since we don't have input controls for

these AR ChIP-seq, we will just run the AR tumor ChIP-seq over the AR normal ChIP-seq (pretend the latter to be input control) to find differential peaks. How many peaks do you get with FDR < 0.01 and fold change > 6?

```
# your bash code here

macs2 callpeak -t q2.tumor.bed -c q2.normal.bed -f AUTO -g hs -q 0.01 --fe-cutoff 6 --outdir q4.out -n

# see peak number of the last peak, shows 10125
tail -n 1 q3.out/compare_peaks.xls
```

There were **10,125 differential peaks** using cuttoffs of FDR < 0.01 and fold change > 6.

Part II. ChIP-seq quality control

Question 5

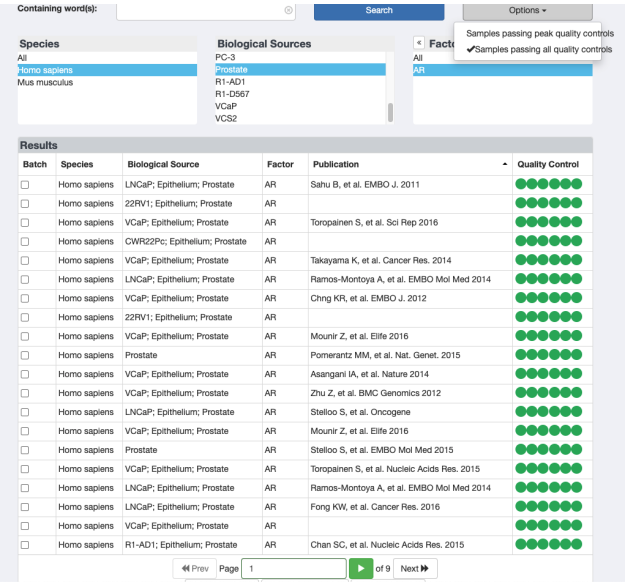
Cistrome Data Browser (http://cistrome.org/db/) has collected and pre-processed a large compendium of the published ChIP-seq data in the public. Play with Cistrome DB. Biological sources indicate whether the ChIP-seq is generated from a cell line (e.g. VCaP, LNCaP, PC3, C4-2) or a tissue (Prostate). Are there over 100 AR ChIP-seq samples which passed all QC meatures in human prostate tissues?

Hint: Check out Options next to the Search function.

I selected species as homo sapeins, biological sources as prostate, factors as AR, and options as samples passing all quality controls.

There are eight full pages of twenty results and a ninth page with one result. There are a total of 161, so **yes, there are over 100 AR ChIP-seq samples which passed all QC measures in human prostate tissues**.

```
knitr::include_graphics("q5.cistrome.browser.png")
```



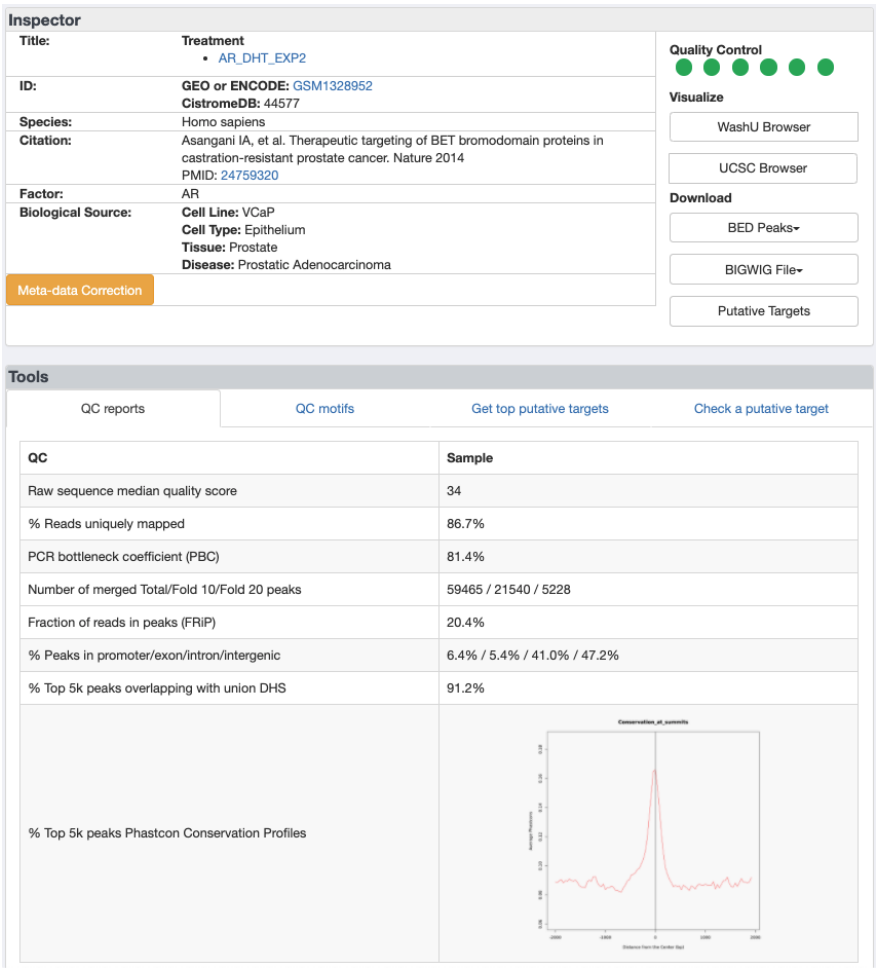
Question 6

Doing transcription factor ChIP-seq in tissues could be a tricky experiment, so sometimes even published data in high profile journals have bad quality. Look at a few AR ChIP-seq samples in the prostate tissue on Cistrome and inspect their QC reports. Can you comment

on what QC measures tell you whether a ChIP-seq is of good or bad quality? Include a screen shot of a good AR ChIP-seq vs a bad AR ChIP-seq.

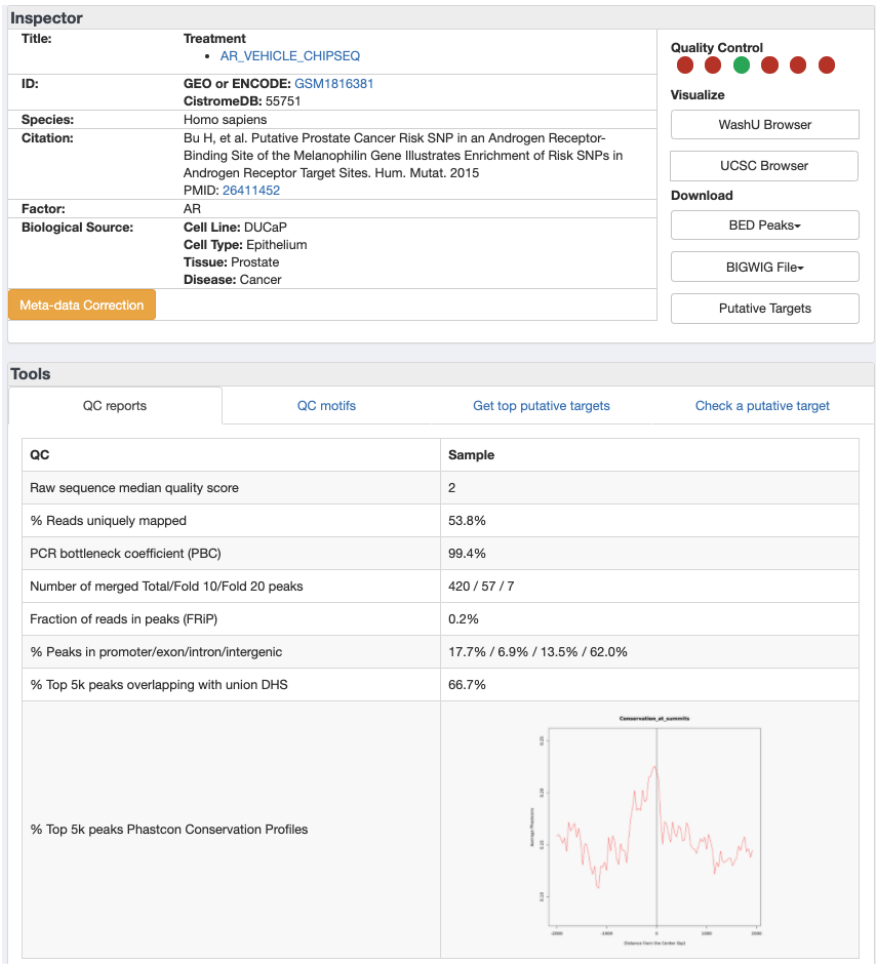
This first graphic is an example of **a good AR ChIP-seq that passes all quality control metrics**, used for the paper by Asangani IA, et al. on Therapeutic targeting of BET bromodomain proteins in castration-resistant prostate cancer.

```
knitr::include_graphics('q6.good.png')
```



This second graphic is an example of **a bad AR ChIP-seq that only passes one quality control metric**, used for the paper by Bu H, et al.

```
knitr::include_graphics('q6.bad.png')
```



It looks like the raw sequence median quality QC metric is the least often to fail in these publications. This metric is the sample's median sequence quality. If this value is below 25, the sequence is considered "bad quality". All QC metrics are important, but this one seems like one that can easly categorize a sequence as bad or good.

Question 7 For Graduate Students Only

Antibody is one important factor influencing the quality of a ChIP-seq experiment. Click on the GEO (GSM) ID of some good quality vs bad quality ChIP-seq data, and see where they got their AR antibodies. If you plan to do an AR ChIP-seq experiment, which company and catalog # would you use to order the AR antibody?

I looked at the AR antibody company and catalog number for a few very good datasets (that pass all QC measures), and for some bad datasets (that fail at least 4). The results are below. I frequently saw **SC-816 from Santa Cruz Biotechnology** used on good datasets, and didn't come across a bad dataset that had used it, so I would probably plan to use that antibody.

Good ones

- Santa Cruz Biotechnology, SC-816 lot B1012
- Millipore, 06-680
- Santa Cruz Biotechnology, sc-816
- Santa Cruz Biotechnology, scbt-816X
- Santa Cruz Biotechnology, SC-816

Bad ones

- Santa Cruz Biotechnology, sc-13062
- Millipore UB 06-680

Part III ChIP-seq motif finding

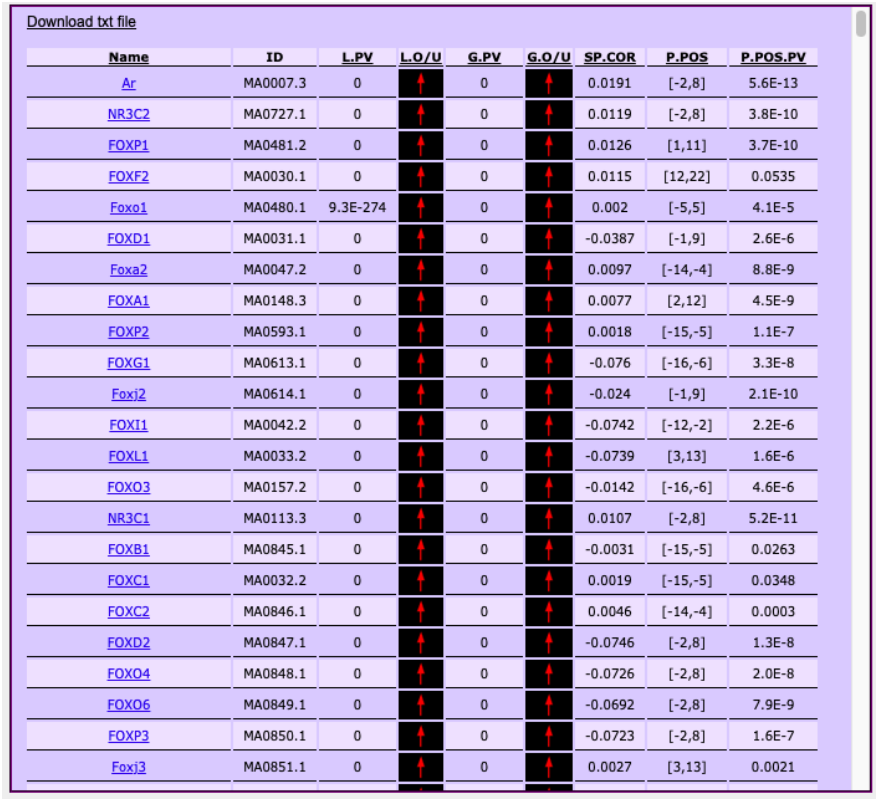
Question 8 For Graduate Students Only

We want to see in prostate tumors, which other transcription factors (TF) might be collaborating with AR. You can try any of the following motif finding tools to find TF motifs enriched in the differential AR peaks you identified above. Did you find the known AR motif, and motifs of other factors that might interact with AR in prostate cancer in gene regulation? Describe the tool you used, what you did, and what you found. Note that finding the correct AR motif is usually an important criterion for AR ChIP-seq QC as well.

HOMER: http://homer.ucsd.edu/homer/motif/
MEME: http://meme-suite.org/tools/meme-chip
Weeder: http://159.149.160.88/pscan_chip_dev/
Cistrome: http://cistrome.org/ap/root (Register a free account).

Using Weeder, I first uploaded the BED file from part I question 4. I selected hg38 for assembly, and clicked run. AR is the top enriched motif within the subset of locations where AR is differentially enriched, which is what we expect to see. The next highest enriched are NR3C2, FOXP1, and FOXF2. There are a lot of "FOX" and "NR" prefixed motifs. The FOX (Forehead box) family is known to be associated with cancers in general and prostate cancer specifically. This is verified in many papers, including "The Dominant Role of Forkhead Box Proteins in Cancer" and "Current perspectives on FOXA1 regulation of androgen receptor signaling and prostate cancer". The NR (nuclear receptor) family has also been studied and has been found to have roles in prostate cancer in papers including "The Role of Nuclear Receptors in Prostate Cancer".

```
knitr::include_graphics("q8.weeder.list.png")
```



Download txt file

Name	ID	L.PV	L.O/U	G.PV	G.O/U	SP.COR	P.POS	P.POS.PV
Ar	MA0007.3	0	↑	0	↑	0.0191	[-2,8]	5.6E-13
NR3C2	MA0727.1	0	↑	0	↑	0.0119	[-2,8]	3.8E-10
FOXP1	MA0481.2	0	↑	0	↑	0.0126	[1,11]	3.7E-10
FOXF2	MA0030.1	0	↑	0	↑	0.0115	[12,22]	0.0535
Foxo1	MA0480.1	9.3E-274	↑	0	↑	0.002	[-5,5]	4.1E-5
FOXD1	MA0031.1	0	↑	0	↑	-0.0387	[-1,9]	2.6E-6
Foxa2	MA0047.2	0	↑	0	↑	0.0097	[-14,-4]	8.8E-9
FOXA1	MA0148.3	0	↑	0	↑	0.0077	[2,12]	4.5E-9
FOXP2	MA0593.1	0	↑	0	↑	0.0018	[-15,-5]	1.1E-7
FOXG1	MA0613.1	0	↑	0	↑	-0.076	[-16,-6]	3.3E-8
Foxj2	MA0614.1	0	↑	0	↑	-0.024	[-1,9]	2.1E-10
FOXI1	MA0042.2	0	↑	0	↑	-0.0742	[-12,-2]	2.2E-6
FOXL1	MA0033.2	0	↑	0	↑	-0.0739	[3,13]	1.6E-6
FOXO3	MA0157.2	0	↑	0	↑	-0.0142	[-16,-6]	4.6E-6
NR3C1	MA0113.3	0	↑	0	↑	0.0107	[-2,8]	5.2E-11
FOXB1	MA0845.1	0	↑	0	↑	-0.0031	[-15,-5]	0.0263
FOXC1	MA0032.2	0	↑	0	↑	0.0019	[-15,-5]	0.0348
FOXC2	MA0846.1	0	↑	0	↑	0.0046	[-14,-4]	0.0003
FOXD2	MA0847.1	0	↑	0	↑	-0.0746	[-2,8]	1.3E-8
FOXO4	MA0848.1	0	↑	0	↑	-0.0726	[-2,8]	2.0E-8
FOXO6	MA0849.1	0	↑	0	↑	-0.0692	[-2,8]	7.9E-9
FOXP3	MA0850.1	0	↑	0	↑	-0.0723	[-2,8]	1.6E-7
Foxj3	MA0851.1	0	↑	0	↑	0.0027	[3,13]	0.0021

Question 9

Look at the AR binding distribution in Cistrome DB from a few good AR ChIP-seq data in prostate. Does AR bind mostly in the gene promoters, exons, introns, or intergenic regions? Also, look at the QC motifs to see what motifs are enriched in the ChIP-seq peaks. Do you see similar motifs here as those you found in your motif analyses?

When looking at a good dataset in Cistrome DB, the QC report "% Peaks in promoter/exon/intron/intergenic" will tell us the distribution of AR binding. I looked at five good (passing all QC) datasets, and found these results:

promoter / exon / intron / intergenic (Paper)

- 2.1% / 3.2% / 44.9% / 49.8% (Mounir Z, et al. Elife 2016)
- 6.4% / 5.4% / 41.0% / 47.2% (Asangani IA, et al. Nature 2014)
- 1.1% / 1.9% / 43.0% / 54.0% (Stelloo S, et al. Oncogene)
- 1.6% / 2.5% / 45.3% / 50.6% (Barfeld SJ, et al. EBioMedicine 2017)
- 1.5% / 2.1% / 44.9% / 51.6% (McNair C, et al. Oncogene 2016)

AR mostly binds in intergenic regions, closely followed by introns. Each of these sections take up nearly half of the distribution, and the remaining AR binding sites are split between promotors and exons.

Part IV. Identify AR-interacting transcription factors

Question 10 For Graduate Students Only

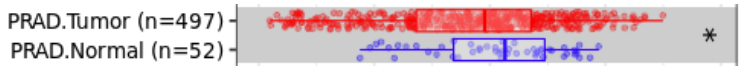
Sometimes members of the same transcription factor family (e.g. E2F1, 2, 3, 4, 5, etc) have similar binding motifs, significant overlapping binding sites (but they might be expressed in very different tissues), and related functions (they could also have different functions if they interact with different partners or compete for binding to the same sites in the same cell). Therefore, to confirm that we have found the correct TFs interacting with AR in prostate tumors, in addition to looking for motifs enriched in the AR ChIP-seq, we also want to see whether the TFs are highly expressed in prostate tumor. For this, we will use the Exploration Component on TIMER (http://timer.cistrome.org/) or GEPIA (http://gepia2.cancer-pku.cn/#general). First, look at differential expression of genes in tumors. Based on the top non-AR motifs you found before, see which member of the TF family that recognizes the motif is highly expressed in prostate tissues or tumors. Another way is to see whether the TF family member and AR have correlated expression pattern in prostate tumors. Enter AR as your interested gene and another gene which is the potential AR collaborator based on the motif, and see whether the candidate TF is correlated with AR in prostate tumors. Based on the motif and expression evidences, which factor in each motif family is the most likely collaborator of AR in prostate cancer?

Note: When we conduct RNA-seq on prostate tumors, each tumor might contain cancer cells, normal prostate epithelia cells, stromal fibroblasts, and other immune cells. Therefore, genes that are highly expressed in cancer cells (including AR) could be correlated in different tumors simply due to the tumor purity bias. Therefore, when looking for genes correlated with AR just in the prostate cancer cells, we should correct this tumor purity bias.

First, I looked at the DE expression of AR across tumor and normal samples. Expression was significantly different at the significance level of 0.05.

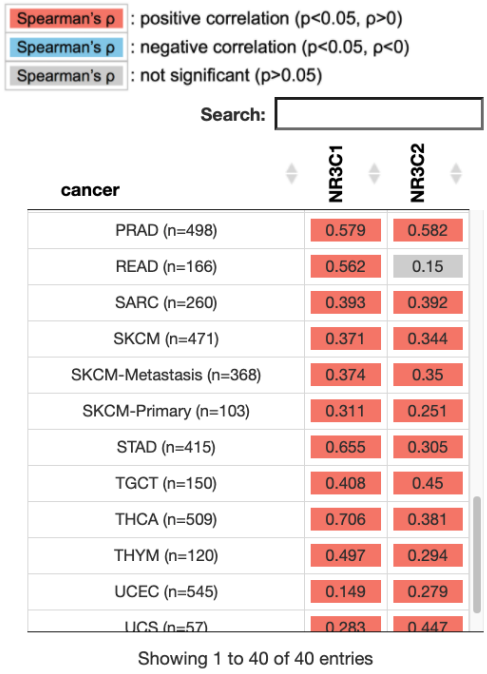
Here the x axis is AR expression level (log2TPM).

```
include_graphics('q10.de.png')
```

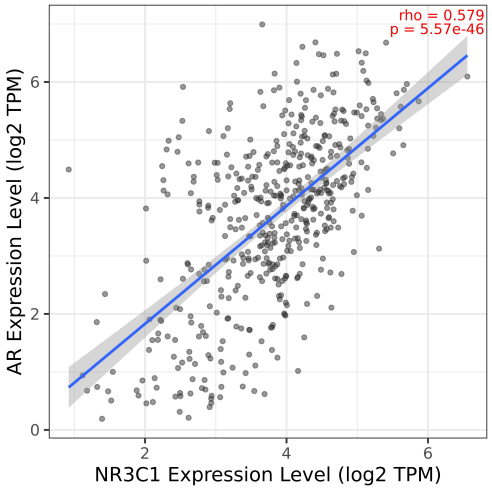


Using TIMER, I looked at correlations between the expression of AR and the NR3C2 and NR3C1 genes (the fox family showed up more frequently, but this was done in lab, and the TA mentioned that we should explore a different family). This table shows that both transcription factors are moderately strongly correlated with AR, with spearman's correlation coefficients of 0.579 for NR3C1 and 0.582 for NR3C2. Detailed plots of these correlations are below: both show relatively linear associations.

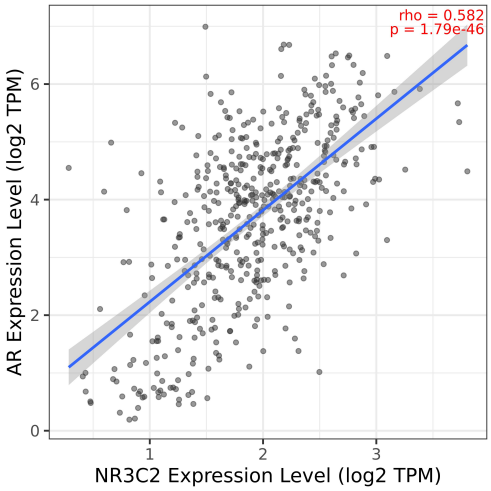
```
include_graphics('q10.table.png')
```




```
include_graphics('q10.plot1.jpg')
```




```
include_graphics('q10.plot2.jpg')
```

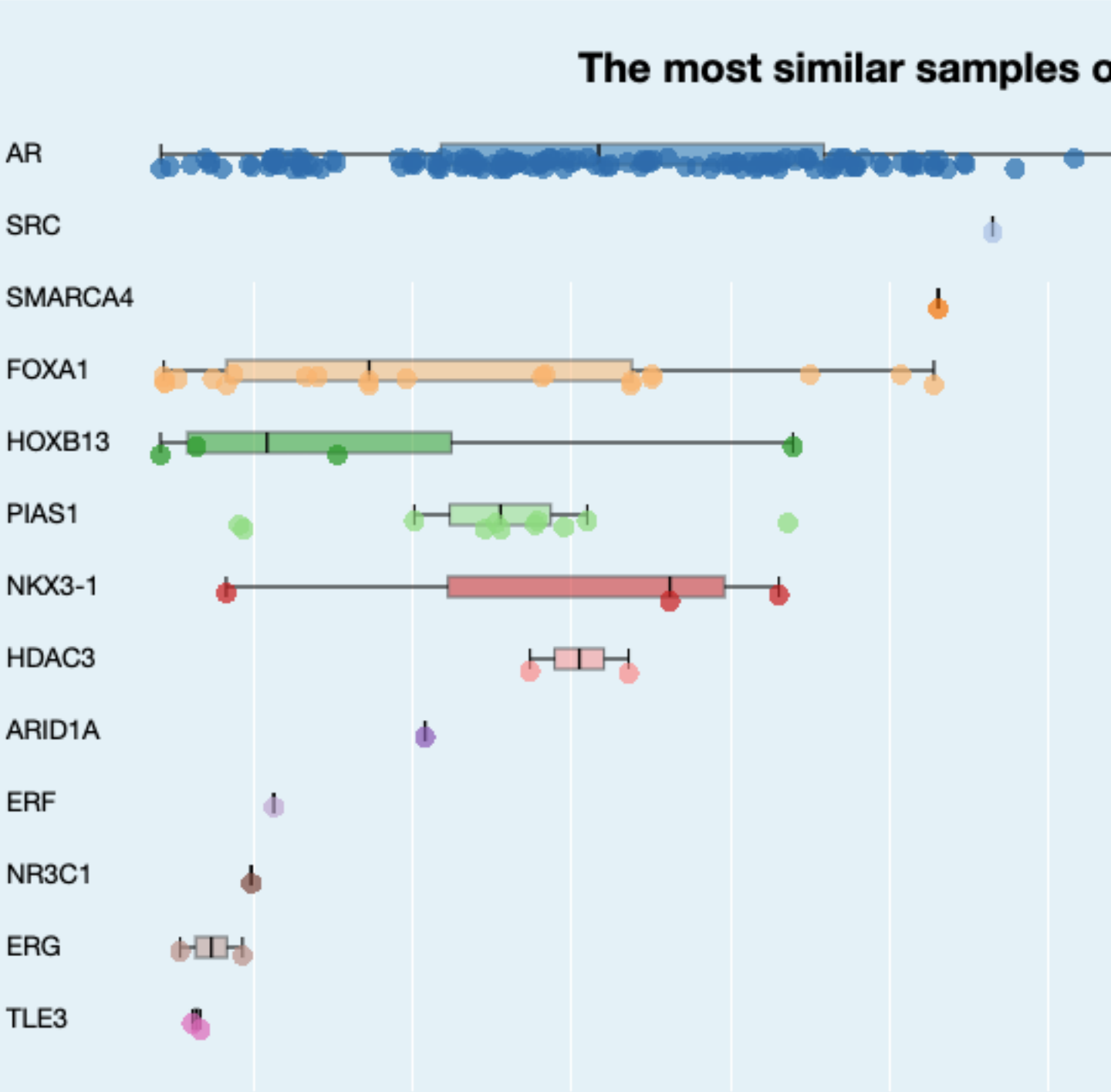


Question 11

Besides looking for motif enrichment, another way to find TFs that might interact with AR is to see whether there are other TF ChIP-seq data which have significant overlap with AR ChIP-seq. Take the differential AR ChIP-seq peaks (in .bed format) between tumor / normal, and run this on the Cistrome Toolkit (http://dbtoolkit.cistrome.org/). The third function in Cistrome Toolkit looks through all the ChIP-seq data in CistromeDB to find ones with significant overlap with your peak list. You should see AR enriched in the results (since your input is a list of AR ChIP-seq peaks after all). What other factors did you see enriched? Do they agree with your motif analyses before?

The transcription factors with the most overlap with AR ChIP-seq are SRC, SMARCA4, FOXA1, and HOXB13. Because the FOX family and NR3C family are represented here, it seems to match my previous results that there are correlations between AR and these families.

```
knitr::include_graphics('q11.plot.png')
```



PART V. Find TF direct target genes and pathways

Question 12 For Graduate Students Only

Now we try to see what direct target genes these AR binding sites regulate. Among the differentially expressed genes in prostate cancer, only a subset might be directly regulated by AR binding. In addition, among all the genes with nearby AR binding, only a subset might be differentially expressed. One simple way of getting the AR target genes is to look at which genes have AR binding in its promoters. Write a python program that takes two input files:

1) the AR differential ChIP-seq peaks in tumor over normal; 2) refGene annotation. The program outputs to a file containing genes that have AR ChIP-seq peak (in this case, stronger peak in tumor) within 2KB +/- from the transcription start site (TSS) of each gene. How many putative AR target genes in prostate cancer do you get using this approach?

Note: From UCSC (http://hgdownload.soe.ucsc.edu/goldenPath/hg38/database/), download the human RefSeq annotation table (find the file refGene.txt.gz for Hg38). To understand the columns in this file, check the query annotation at http://hgdownload.soe.ucsc.edu/goldenPath/hg38/database/refGene.sql.

Hint: 1) The RefGene annotation table is already in **/n/stat115/2021/HW3/refGene.txt** in Cannon. 2) TSS is different for genes on positive or negative strand, i.e. TSS is "txStart" for genes on the positive strand, "txEnd" for genes in negative strand. When testing your python code, try smaller number of gene annotations or smaller number of peaks to check your results before moving forward. 3) Instead of writing the whole process in Python code (which might take a long time to run), you can rewrite TSS starting positions of refGene based on strand in Python, and then perform the 2KB +/- check by command line tool BEDTools (https://bedtools.readthedocs.io/en/latest/) . your answer

```
import pandas as pd
TSS = pd.read_table('refGene.txt')

TSS_sub = TSS.copy().iloc[:,[2,4,5,3,12]]
TSS_sub.columns = ['chr','start','end','strand','id']

TSS_pos = TSS_sub[TSS_sub.strand == '+']
TSS_neg = TSS_sub[TSS_sub.strand == '-']

TSS_pos.end = TSS_pos.start
TSS_neg.start = TSS_neg.end
TSS_new = TSS_pos.append(TSS_neg)
TSS_new.end = TSS_new.end + 1

TSS_new.to_csv("TSS_hg38_bed_file", sep = '\t', header = False, index = False)
```

```
module load centos6/0.0.1-fasrc01
module load bedtools/2.17.0-fasrc01

bedtools window -w 2000 -u -a TSS_hg38_bed_file -b q4.out/compare_summits.bed > q12.ar.diff.bed

cat q12.ar.diff.bed | cut -f5 | sort | uniq | wc -l # shows 404
```

```
# confirm results in R
out <- read.table('q12.ar.diff.bed')
length(unique(out$V5)) # shows 404
```

```
## [1] 404
```

Using this approach, I find **404 unique putative AR target genes in prostate cancer**.

Question 13 For Graduate Students Only

Now overlap the putative AR target genes you get from above with up regulated genes in prostate cances. Try to run GO (e.g. DAVID, you can try other ones too) analysis on 1) the AR target genes from binding alone and 2) the AR target genes by overlapping AR binding with differential expression. Are there enriched GO terms or pathways?

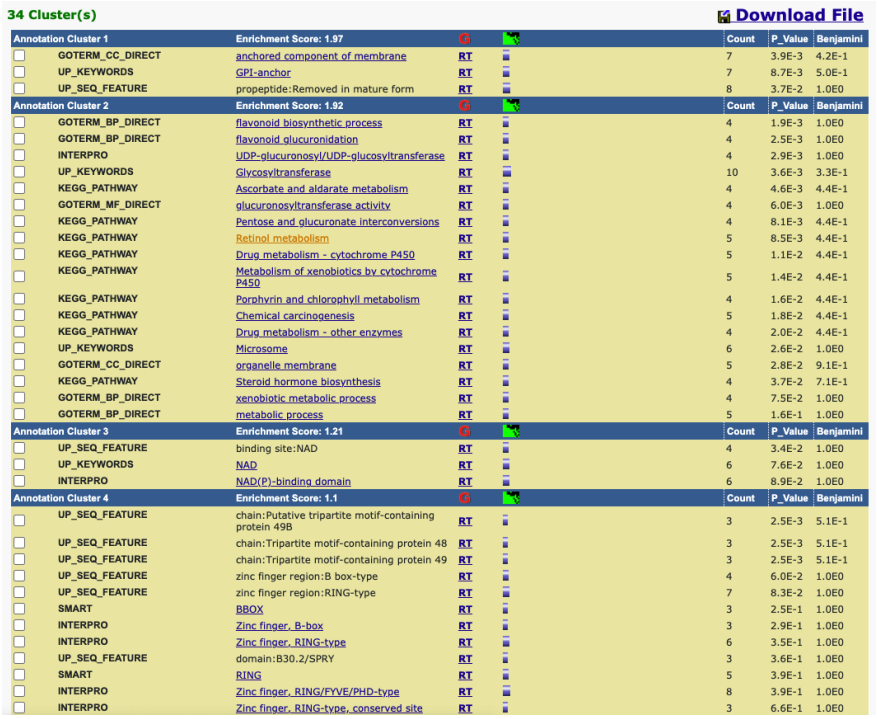
Hint: We have pre-computed the up-regulated genes by comparing a large number of protate tumors with normal prostates, and the results are in **/n/stat115/2021/HW3/up_regulated_genes_in_prostate_cancer.txt** in Cannon.

```
from_binding <- read.table('q12.ar.diff.bed')
upregulated <- read.table('up_regulated_genes_in_prostate_cancer.txt', header = 1)
overlap <- intersect(upregulated$geneName, from_binding$V5)

# cat(paste(unique(from_binding$V5), collapse = '\n'))
# cat(paste(overlap, collapse = '\n')) # print out to copy and paste list

# length(overlap) # shows 12
```

First, I ran DAVID on the unique 404 genes found as putative AR target genes. Of these, 12 unique genes were overlapped between AR binding and differential expression, and I ran DAVID on these overlapped genes too.

Functional Annotation Clusters of Putative AR Genes

```
knitr::include_graphics('q13.from_binding.clusters.png')
```



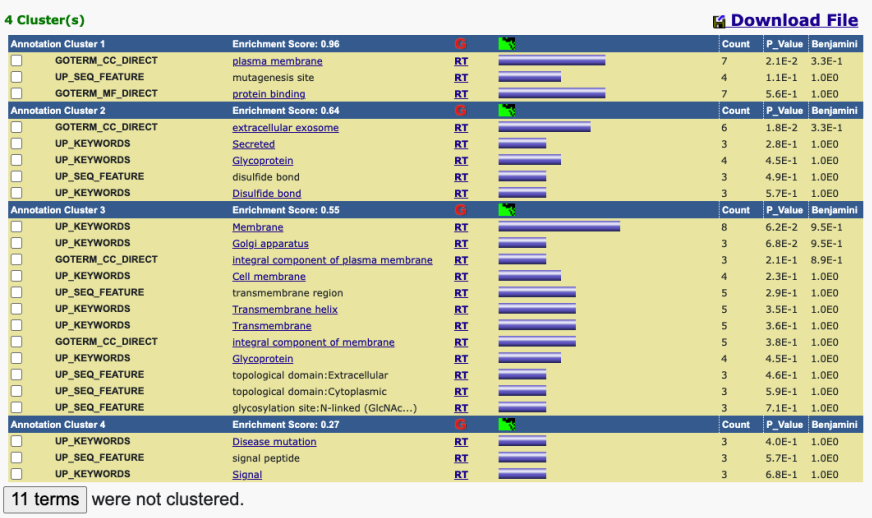
34 Cluster(s) — Download File

Annotation Cluster 1	**Enrichment Score: 1.97**			**Count**	**P_Value**	**Benjamini**
	GOTERM_CC_DIRECT	anchored component of membrane	RT		7	3.9E-3	4.2E-1
	UP_KEYWORDS	GPI-anchor	RT		7	8.7E-3	5.0E-1
	UP_SEQ_FEATURE	propeptide:Removed in mature form	RT		8	3.7E-2	1.0E0
Annotation Cluster 2	**Enrichment Score: 1.92**			**Count**	**P_Value**	**Benjamini**
	GOTERM_BP_DIRECT	flavonoid biosynthetic process	RT		4	1.9E-3	1.0E0
	GOTERM_BP_DIRECT	flavonoid glucuronidation	RT		4	2.5E-3	1.0E0
	INTERPRO	UDP-glucuronosyl/UDP-glucosyltransferase	RT		4	2.9E-3	1.0E0
	UP_KEYWORDS	Glycosyltransferase	RT		10	3.6E-3	3.3E-1
	KEGG_PATHWAY	Ascorbate and aldarate metabolism	RT		4	4.6E-3	4.4E-1
	GOTERM_MF_DIRECT	glucuronosyltransferase activity	RT		4	6.0E-3	1.0E0
	KEGG_PATHWAY	Pentose and glucuronate interconversions	RT		4	8.1E-3	4.4E-1
	KEGG_PATHWAY	Retinol metabolism	RT		5	8.5E-3	4.4E-1
	KEGG_PATHWAY	Drug metabolism - cytochrome P450	RT		5	1.1E-2	4.4E-1
	KEGG_PATHWAY	Metabolism of xenobiotics by cytochrome P450	RT		5	1.4E-2	4.4E-1
	KEGG_PATHWAY	Porphyrin and chlorophyll metabolism	RT		4	1.6E-2	4.4E-1
	KEGG_PATHWAY	Chemical carcinogenesis	RT		5	1.8E-2	4.4E-1
	KEGG_PATHWAY	Drug metabolism - other enzymes	RT		4	2.0E-2	4.4E-1
	UP_KEYWORDS	Microsome	RT		6	2.6E-2	1.0E0
	GOTERM_CC_DIRECT	organelle membrane	RT		5	2.8E-2	9.1E-1
	KEGG_PATHWAY	Steroid hormone biosynthesis	RT		4	3.7E-2	7.1E-1
	GOTERM_BP_DIRECT	xenobiotic metabolic process	RT		4	7.5E-2	1.0E0
	GOTERM_BP_DIRECT	metabolic process	RT		5	1.6E-1	1.0E0
Annotation Cluster 3	**Enrichment Score: 1.21**			**Count**	**P_Value**	**Benjamini**
	UP_SEQ_FEATURE	binding site:NAD	RT		4	3.4E-2	1.0E0
	UP_KEYWORDS	NAD	RT		6	7.6E-2	1.0E0
	INTERPRO	NAD(P)-binding domain	RT		6	8.9E-2	1.0E0
Annotation Cluster 4	**Enrichment Score: 1.1**			**Count**	**P_Value**	**Benjamini**
	UP_SEQ_FEATURE	chain:Putative tripartite motif-containing protein 49B	RT		3	2.5E-3	5.1E-1
	UP_SEQ_FEATURE	chain:Tripartite motif-containing protein 48	RT		3	2.5E-3	5.1E-1
	UP_SEQ_FEATURE	chain:Tripartite motif-containing protein 49	RT		3	2.5E-3	5.1E-1
	UP_SEQ_FEATURE	zinc finger region:B box-type	RT		4	6.0E-2	1.0E0
	UP_SEQ_FEATURE	zinc finger region:RING-type	RT		7	8.3E-2	1.0E0
	SMART	BBOX	RT		3	2.5E-1	1.0E0
	INTERPRO	Zinc finger, B-box	RT		3	2.9E-1	1.0E0
	INTERPRO	Zinc finger, RING-type	RT		6	3.5E-1	1.0E0
	UP_SEQ_FEATURE	domain:B30.2/SPRY	RT		3	3.6E-1	1.0E0
	SMART	RING	RT		5	3.9E-1	1.0E0
	INTERPRO	Zinc finger, RING/FYVE/PHD-type	RT		8	3.9E-1	1.0E0
	INTERPRO	Zinc finger, RING-type, conserved site	RT		3	6.6E-1	1.0E0

The 404 AR target genes from binding alone result in 34 clusters. However, Benjamini Hochberg adjusted p-values are insignificant for all terms. Because of this insignificance, **there are not enriched terms or pathways**.

Functional Annotation Clusters of DE Putative AR Genes

```
knitr::include_graphics('q13.overlap.clusters.png')
```



4 Cluster(s) — Download File

Annotation Cluster 1	**Enrichment Score: 0.96**			**Count**	**P_Value**	**Benjamini**
GOTERM_CC_DIRECT	plasma membrane	RT		7	2.1E-2	3.3E-1
UP_SEQ_FEATURE	mutagenesis site	RT		4	1.1E-1	1.0E0
GOTERM_MF_DIRECT	protein binding	RT		7	5.6E-1	1.0E0
Annotation Cluster 2	**Enrichment Score: 0.64**			**Count**	**P_Value**	**Benjamini**
GOTERM_CC_DIRECT	extracellular exosome	RT		6	1.8E-2	3.3E-1
UP_KEYWORDS	Secreted	RT		3	2.8E-1	1.0E0
UP_KEYWORDS	Glycoprotein	RT		4	4.5E-1	1.0E0
UP_SEQ_FEATURE	disulfide bond	RT		3	4.9E-1	1.0E0
UP_KEYWORDS	Disulfide bond	RT		3	5.7E-1	1.0E0
Annotation Cluster 3	**Enrichment Score: 0.55**			**Count**	**P_Value**	**Benjamini**
UP_KEYWORDS	Membrane	RT		8	6.2E-2	9.5E-1
UP_KEYWORDS	Golgi apparatus	RT		3	6.8E-2	9.5E-1
GOTERM_CC_DIRECT	integral component of plasma membrane	RT		3	2.1E-1	8.9E-1
UP_KEYWORDS	Cell membrane	RT		4	2.3E-1	1.0E0
UP_SEQ_FEATURE	transmembrane region	RT		5	2.9E-1	1.0E0
UP_KEYWORDS	Transmembrane helix	RT		5	3.5E-1	1.0E0
UP_KEYWORDS	Transmembrane	RT		5	3.6E-1	1.0E0
GOTERM_CC_DIRECT	integral component of membrane	RT		5	3.8E-1	1.0E0
UP_KEYWORDS	Glycoprotein	RT		4	4.5E-1	1.0E0
UP_SEQ_FEATURE	topological domain:Extracellular	RT		3	4.6E-1	1.0E0
UP_SEQ_FEATURE	topological domain:Cytoplasmic	RT		3	5.9E-1	1.0E0
UP_SEQ_FEATURE	glycosylation site:N-linked (GlcNAc...)	RT		3	7.1E-1	1.0E0
Annotation Cluster 4	**Enrichment Score: 0.27**			**Count**	**P_Value**	**Benjamini**
UP_KEYWORDS	Disease mutation	RT		3	4.0E-1	1.0E0
UP_SEQ_FEATURE	signal peptide	RT		3	5.7E-1	1.0E0
UP_KEYWORDS	Signal	RT		3	6.8E-1	1.0E0

11 terms were not clustered.

The 12 Differentially Expressed AR target genes from binding alone result in 4 clusters. However, again, Benjamini Hochberg adjusted p-values are insignificant for all terms. Because of this insignificance, **there are not enriched terms or pathways**.

Question 14

Another way of getting the AR target genes is to consider the number of AR binding sites within 100KB of TSS, but weight each binding site by an exponential decay of its distance to the gene TSS (i.e. peaks closer to TSS have higher weights). For this, we have calculated regulatory potential score for each refseq gene. Select the top 1500 genes with highest regulatory potential score, try to run GO analysis both with and without differential expression (overlap with the up-regulated genes), and see the enriched GO terms.

Hints: 1). We have pre-computed the regulatory potential for each gene based on the differential AR peaks and put this in **/n/stat115/2021/HW3/AR_peaks_regulatory_potential.txt** in Cannon. 2) Basically this approach assumes that there are stronger AR targets (e.g. those genes with many AR binding sites within 100KB and have stronger differential expression) and weaker AR targets, instead of a binary Yes / No AR targets.

your answer

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
regulatory <- read.table('AR_peaks_regulatory_potential.txt')
regulatory %>%
  arrange(-V5) %>%
  pull(V7) %>%
  unique() %>%
  head(1500) %>%
```

```
  paste(sep = '\n') %>%
  write.csv(file = "q14.top1500.txt", quote = FALSE, row.names = FALSE)

regulatory %>%
  arrange(-V5) %>%
  pull(V7) %>%
  unique() %>%
  head(1500) %>%
  intersect(upregulated$geneName) %>%
  paste(sep = '\n') %>%
  write.csv(file = 'q14.intersect.txt', quote = F, row.names = F)
```

Clusters for Top 1500 Regulatory Potential

```
knitr::include_graphics("q14.top1500.png")
```

150 Cluster(s)

Annotation Cluster 1	Enrichment Score: 2.51	G	
UP_KEYWORDS	Membrane	RT	
UP_KEYWORDS	Cell membrane	RT	
GOTERM_CC_DIRECT	plasma membrane	RT	
Annotation Cluster 2	**Enrichment Score: 2.39**	**G**	
KEGG_PATHWAY	Pentose and glucuronate interconversions	RT	
KEGG_PATHWAY	Ascorbate and aldarate metabolism	RT	
GOTERM_BP_DIRECT	flavonoid biosynthetic process	RT	
INTERPRO	UDP-glucuronosyl/UDP-glucosyltransferase	RT	
GOTERM_BP_DIRECT	flavonoid glucuronidation	RT	
KEGG_PATHWAY	Porphyrin and chlorophyll metabolism	RT	
KEGG_PATHWAY	Retinol metabolism	RT	
KEGG_PATHWAY	Drug metabolism - other enzymes	RT	
UP_KEYWORDS	Microsome	RT	
GOTERM_MF_DIRECT	glucuronosyltransferase activity	RT	
KEGG_PATHWAY	Steroid hormone biosynthesis	RT	
KEGG_PATHWAY	Metabolism of xenobiotics by cytochrome P450	RT	
KEGG_PATHWAY	Drug metabolism - cytochrome P450	RT	
KEGG_PATHWAY	Chemical carcinogenesis	RT	
GOTERM_MF_DIRECT	transferase activity, transferring hexosyl groups	RT	
GOTERM_BP_DIRECT	cellular glucuronidation	RT	
GOTERM_BP_DIRECT	xenobiotic metabolic process	RT	
GOTERM_CC_DIRECT	organelle membrane	RT	
GOTERM_MF_DIRECT	retinoic acid binding	RT	
Annotation Cluster 3	**Enrichment Score: 2.09**	**G**	
GOTERM_MF_DIRECT	sialyltransferase activity	RT	
INTERPRO	Glycosyl transferase, family 29	RT	
GOTERM_BP_DIRECT	sialylation	RT	
GOTERM_BP_DIRECT	oligosaccharide metabolic process	RT	
GOTERM_BP_DIRECT	ganglioside biosynthetic process	RT	
GOTERM_MF_DIRECT	alpha-N-acetylgalactosaminide alpha-2,6-sialyltransferase activity	RT	
KEGG_PATHWAY	Glycosphingolipid biosynthesis - ganglio series	RT	
GOTERM_BP_DIRECT	protein N-linked glycosylation via asparagine	RT	
GOTERM_BP_DIRECT	protein glycosylation	RT	
INTERPRO	Sialyltransferase	RT	
PIR_SUPERFAMILY	sialyltransferase	RT	
GOTERM_CC_DIRECT	integral component of Golgi membrane	RT	

Looking at the top 1500 unique genes with highest regulatory potentials, DAVID resulted in 150 clusters. The only terms that are significant are “Membrane” in cluster one, and “Pentose and glucuronate interconversions” and “Ascorbate and aldarate metabolism” in cluster two.

Clusters for DE Top Regulatory Potential

```
knitr::include_graphics("q14.intersect.png")
```

8 Cluster(s)

Annotation Cluster 1		Enrichment Score: 2.53	G	
	GOTERM_BP_DIRECT	androgen receptor signaling pathway	RT	
	GOTERM_MF_DIRECT	ligand-dependent nuclear receptor transcription coactivator activity	RT	
	GOTERM_MF_DIRECT	RNA polymerase II transcription cofactor activity	RT	
	GOTERM_BP_DIRECT	positive regulation of transcription, DNA-templated	RT	
	UP_KEYWORDS	Activator	RT	
	GOTERM_MF_DIRECT	vitamin D receptor binding	RT	
	GOTERM_MF_DIRECT	transcription cofactor activity	RT	
	GOTERM_BP_DIRECT	intracellular steroid hormone receptor signaling pathway	RT	
	GOTERM_MF_DIRECT	transcription coactivator activity	RT	
	GOTERM_BP_DIRECT	cellular response to estradiol stimulus	RT	
	GOTERM_CC_DIRECT	mediator complex	RT	
	GOTERM_MF_DIRECT	androgen receptor binding	RT	
	GOTERM_BP_DIRECT	transcription initiation from RNA polymerase II promoter	RT	
	KEGG_PATHWAY	Thyroid hormone signaling pathway	RT	
	GOTERM_CC_DIRECT	nucleoplasm	RT	
	GOTERM_BP_DIRECT	positive regulation of transcription from RNA polymerase II promoter	RT	
	UP_KEYWORDS	Transcription regulation	RT	
	UP_KEYWORDS	Transcription	RT	
	UP_KEYWORDS	Nucleus	RT	
	GOTERM_MF_DIRECT	receptor activity	RT	
	GOTERM_CC_DIRECT	nucleus	RT	
	GOTERM_MF_DIRECT	sequence-specific DNA binding	RT	
	KEGG_PATHWAY	HTLV-I infection	RT	
	GOTERM_BP_DIRECT	negative regulation of transcription from RNA polymerase II promoter	RT	
	GOTERM_BP_DIRECT	transcription, DNA-templated	RT	
	UP_KEYWORDS	Receptor	RT	
	GOTERM_BP_DIRECT	regulation of transcription, DNA-templated	RT	
	UP_KEYWORDS	DNA-binding	RT	
Annotation Cluster 2		**Enrichment Score: 1.25**	G	
	GOTERM_BP_DIRECT	oxidation-reduction process	RT	
	UP_KEYWORDS	NAD	RT	
	INTERPRO	NAD(P)-binding domain	RT	
	UP_KEYWORDS	Oxidoreductase	RT	

Looking at the intersection of the top 1500 genes from before and genes that are known to be up-regulated in prostate cancer, DAVID returned 8 clusters. Only the first cluster includes significant terms after multiple hypothesis testing adjustment. The significant terms are related to steroid and other hormone pathways as well as transcription events. Many of these make sense to be enriched. For example, the American Cancer Society writes, "androgens stimulate prostate cancer cells to grow" (link), so having "androgen receptor signaling pathway" near the top of this list is expected.

Question 15 For Graduate Students Only

Comment on the AR targets you get from promoter binding (your code) and distance weighted binding. Which one gives you better function / pathway enrichment? Does considering differential expression help?

The AR targets from promoter binding had no singiciantly enriched terms or pathways, while the distance weighted binding had a few terms show up. Considering differential expresson improved the results from the distance weighted binding because there are more significantly enriched terms known to be associated with prostate cancer. Considering differential expression may also improve the results from promoter binding, but in this example there were still no singificantly enriched terms. However, the p-values were much lower, so it would likely improve these results in another scenario.

Question 16

For what you did in Q12-15, Cistrome-GO (http://go.cistrome.org/) already provides a very simple solution. It performs functional enrichment analysis using a ChIP-seq peak file and an optional differential expression analysis file. Without differential expression, Cistrome-GO will perform GO analysis only on the regulatory potential. Now, use the differential peaks and upregulated genes to run Cistrome-GO and see the enriched biological functions or pathways.

Hint: Please refer to https://academic.oup.com/nar/article/47/W1/W206/5485528

I uploaded the output from question 4 as my Peak Bed File and `up_regulated_genes_in_prostate_cancer.txt` as my differential expression file.

```
knitr::include_graphics('q16.png')
```

KEGG pathways	N, B, n, b	Enrichment	P-value	FDR	Genes
Pentose and glucuronate interconversions	7753, 821, 33, 18	5.151	8.409E-09	2.657E-06	33
Ascorbate and aldarate metabolism	7753, 1093, 27, 17	4.466	8.512E-08	1.345E-05	27
Porphyrin and chlorophyll metabolism	7753, 1093, 42, 20	3.378	2.901E-06	3.056E-04	42
Drug metabolism - cytochrome P450	7753, 1290, 70, 29	2.49	1.237E-05	9.771E-04	70
Metabolic pathways	7753, 1548, 1248, 311	1.248	5.591E-05	3.533E-03	1248
Drug metabolism - other enzymes	7753, 1300, 79, 30	2.265	7.834E-05	4.126E-03	79
Steroid hormone biosynthesis	7753, 990, 60, 21	2.741	1.111E-04	5.016E-03	60
Metabolism of xenobiotics by cytochrome P450	7753, 1290, 74, 28	2.274	1.396E-04	5.514E-03	74
Retinol metabolism	7753, 128, 67, 8	7.232	1.854E-04	6.508E-03	67
Chemical carcinogenesis	7753, 1290, 82, 29	2.126	4.127E-04	1.304E-02	82
Fatty acid metabolism	7753, 1225, 53, 20	2.388	1.015E-03	2.803E-02	53
Thyroid hormone signaling pathway	7753, 879, 116, 28	2.129	1.064E-03	2.803E-02	116
AMPK signaling pathway	7753, 1295, 120, 37	1.846	1.254E-03	3.049E-02	120
Adipocytokine signaling pathway	7753, 1238, 69, 23	2.088	3.237E-03	7.305E-02	69
cAMP signaling pathway	7753, 1497, 212, 62	1.515	4.189E-03	8.825E-02	212
One carbon pool by folate	7753, 1601, 20, 11	2.663	4.968E-03	9.812E-02	20
Glycosphingolipid biosynthesis - ganglio series	7753, 663, 15, 6	4.678	5.801E-03	1.078E-01	15
Fatty acid biosynthesis	7753, 1188, 13, 7	3.514	7.579E-03	1.264E-01	13
Tryptophan metabolism	7753, 1889, 42, 20	1.954	7.603E-03	1.264E-01	42
Amino sugar and nucleotide sugar metabolism	7753, 1409, 47, 18	2.107	8.196E-03	1.295E-01	47
Prostate cancer	7753, 1625, 97, 34	1.672	9.835E-03	1.472E-01	97
Peroxisome	7753, 72, 83, 5	6.487	1.052E-02	1.472E-01	83
Longevity regulating pathway	7753, 1295, 89, 27	1.816	1.071E-02	1.472E-01	89
Oxytocin signaling pathway	7753, 372, 153, 17	2.316	1.254E-02	1.651E-01	153
Apelin signaling pathway	7753, 1189, 137, 35	1.666	1.456E-02	1.840E-01	137

Here we see some KEGG pathways significantly enriched. Some, like pentose and glucuronate interconversions and steroid hormone biosynthesis, we have seen before using other methods.

PART VI. ATAC-seq

ATAC-seq, or Assay for Transposase-Accessible Chromatin coupled with next-gen sequencing, is a technique to characterize accessible chromatin regions in the genome. Suppose the molecular mechanism of prostate cancer development was poorly understood and we didn't know AR was important, then we would not know to do AR ChIP-seq to start with. In this case, ATAC-seq oculd be performed on the prostate cancer tissue without prior knowledge, and we could find the cancer drivers based on the ATAC-seq peaks.

Unlike ChIP-seq which often uses chromatin input as controls, ATAC-seq has no control samples. MACS2 is suited for calling differential ATAC-seq peaks between tumor and normal (similar to your AR differential peak calling in Q4). A second way is to first call the union peaks by combining reads from all the tumor and normal samples, the extract the read counts from each samples in the union peaks, and run DESeq2 to find differential peaks (as if each union peak is a gene in regular DESeq2 analysis). A third way is to call peaks in the tumor, then call peaks in the normal, and just simply do an overlap of the peaks between the two to find different peaks. SAMTools (http://samtools.sourceforge.net/) and BEDTools (https://bedtools.readthedocs.io/en/latest/) are extremely useful tools to manipulate SAM/BAM and BED files.

Question 18

We have found some published ATAC-seq data on prostate tumor and normal prostate tissues, and have done read mapping and the peak calls (MACS2) on each data seaprately. Use BEDTools to find peaks that are unique in the tumor and not in the normal (the 3rd approach). How many tumor-specific peaks (peaks in tumor but not in normal) do you get this way? Run this through Cistrome Toolkit and examine the results. What transcription fators are important in driving these prostate cancer-specific signals?

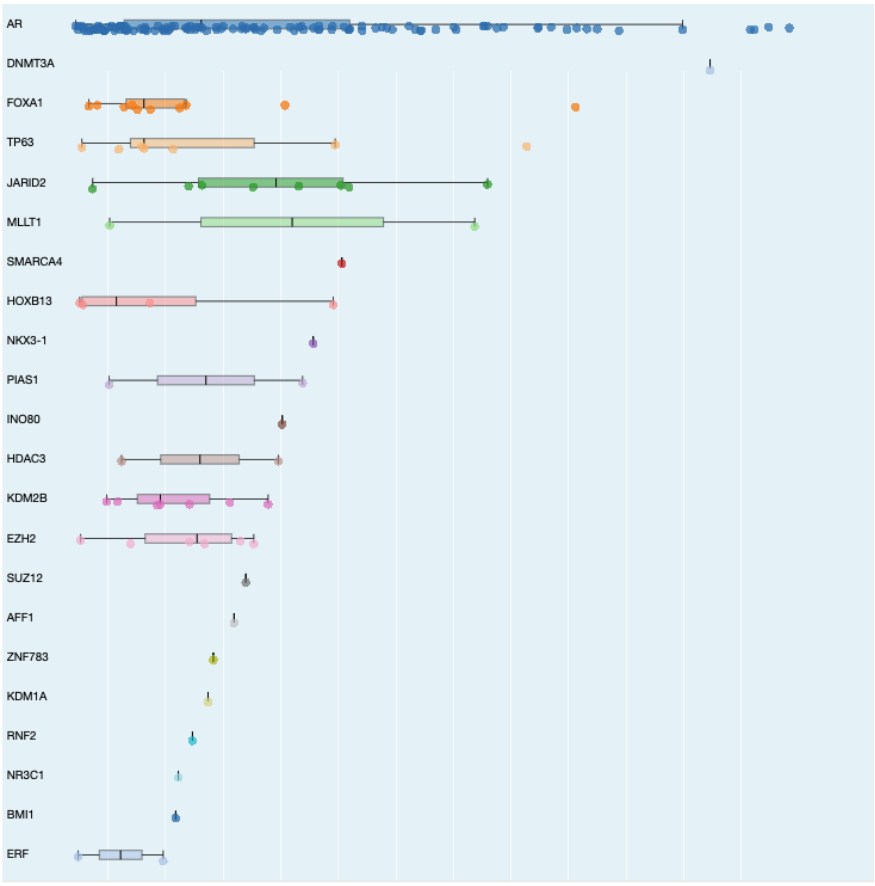
Hint: The peak files are **/n/stat115/2021/HW3/tumor_ATAC_peaks.bed** and **/n/stat115/2021/HW3/normal_A** in Cannon

```
module load centos6/0.0.1-fasrc01
module load bedtools/2.17.0-fasrc01

bedtools subtract -A -a /n/stat115/2021/HW3/tumor_ATAC_peaks.bed -b /n/stat115/2021/HW3/normal_ATAC_peal

cat q18.out.bed | wc -l # shows 13448
```

I used the tumor peaks file as "a" and the normal peak files as "b", so the output will be peaks that are in tumor and not the normal. There are **13,448 tumor-specific peaks**.

I did not include a differential expression file this time, and ran the results of bedtools subtraction through the cistrome toolkit.

```
knitr::include_graphics("q18.png")
```



The transcription fators that are most important in driving these prostate cancer-specific signals are those most similar to the differential peak set in the plot above. In order, the top five are AR, DNMT3A, FOXA1, TP63, and JARID2. Investigating these points further, I found that AR, HOXB13, SMARCA4, and NKX3-1, among others, have the prostate as their biological resource.

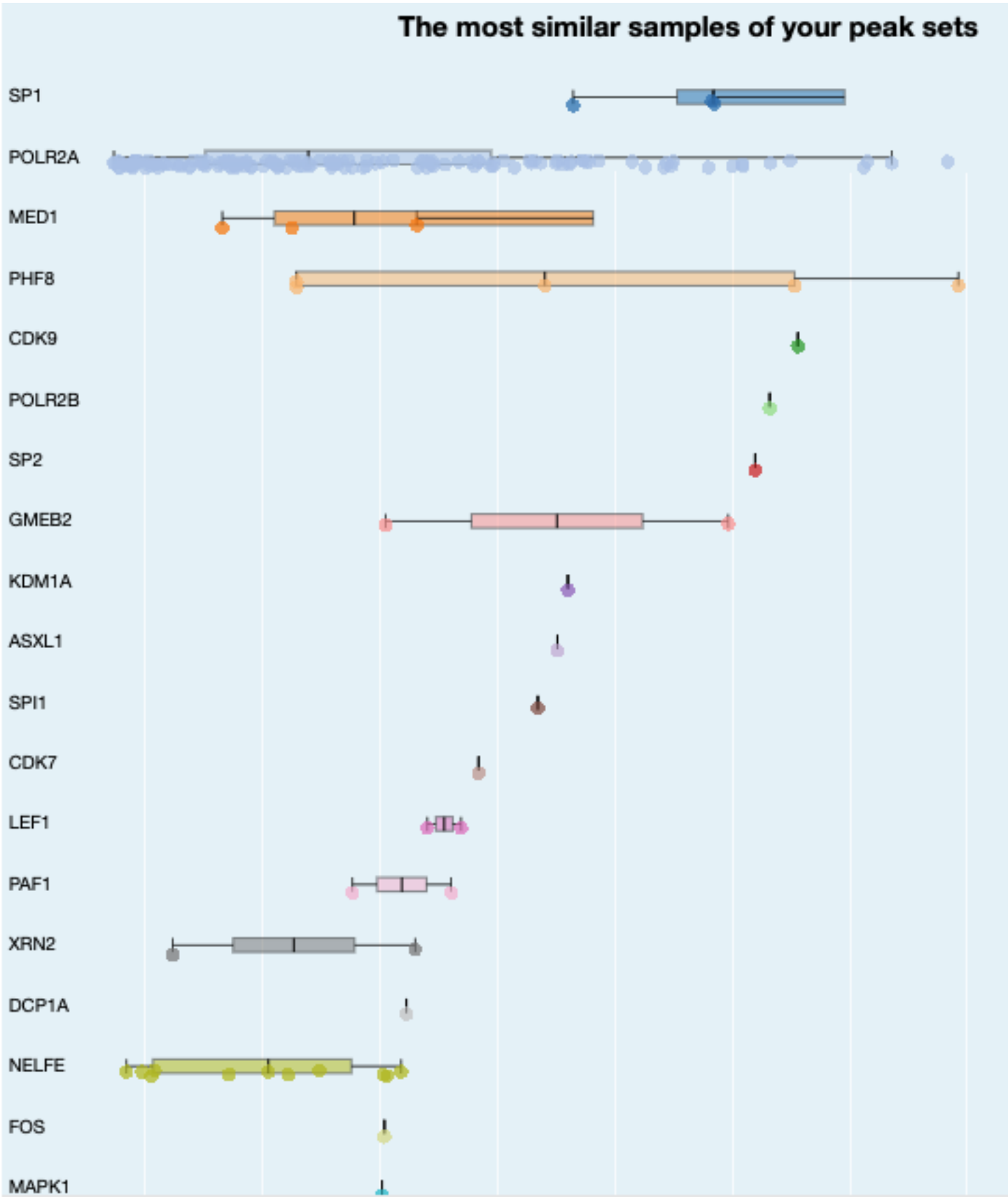
Question 19 For Graduate Students Only

Now just take the top 10K ATAC-seq peaks (ranked by fold-change since these are all significant peaks already) in the prostate tumor only (not the differential peaks) and run Cistrome Toolkit. Compare the results with Q18. Can you comment on which approach gives you more meaning results?

```
tumor <- read.table('tumor_ATAC_peaks.bed')
tumor %>%
  arrange(-V5) %>%
  head(10000) %>%
  write.table('tumor_ATAC_peaks_top10k.bed', sep = '\t', quote = FALSE, row.names = F)
```

I used the tumor peaks file this time. I was going to select the option in Cistrome to use the top 10k peaks, but the tumor peaks file is too large to upload. Instead, I filtered peaks according to enrichment in R and saved the top 10,000 to a smaller `bed` file and uploaded that to Cistrome.

```
knitr::include_graphics("q19.png")
```


The most similar samples of your peak sets
SP1
POLR2A
MED1
PHF8
CDK9
POLR2B
SP2
GMEB2
KDM1A
ASXL1
SPI1
CDK7
LEF1
PAF1
XRN2
DCP1A
NELFE
FOS
MAPK1

There's no overlap in the top few TFs between Questions 18 and 19. The TFs in Question 18 are more widely known to be associated with prostate cancer.

This plot shows similar TFs to this peak set. Because this peak set includes peaks specific to tumors, and peaks that would be present even if the individual did not have cancer, this peak set is relatively meaningless. There is no way to know if these TFs drive prostate cancer-specific signals or if they're normal to have.

On the other hand, the peak set in Question 18 represents peaks that are present in cancer but not in normal tissue, giving more meaning to its results.

Question 20

Sometimes even without ATAC-seq, we can predict the driving transcription factors based on deferentially expressed genes by using public ChIP-seq data. Even if the public TF ChIP-seq data was generated on different cells, they still provide some insights on the TF putative targets. Based on the differential gene expression list, Lisa first tries to build an epigenetic model using public ChIP-seq and chromatin accessibility profiles, then uses public ChIP-seq data to see which TF ChIP-seq fits the chromatin model the best. Now run the up-regulated gene in prostate cancer on Lisa, and see what transcription factors were predicted as putative drivers for these differential genes?

Hint: 1). Please refer to https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-1934-6. 2). You can use Lisa on http://lisa.cistrome.org/, but it was not designed for a hundred students to submit jobs in one day. We have a newer command line version Lisa2 (https://github.com/liulab-dfci/lisa2), which is installed on Cannon and might save you time from waiting for the web server (actually Lisa2 also runs much faster).

your answer

```
module load Anaconda/5.0.1-fasrc02

source /n/stat115/2021/HW5/miniconda3/bin/activate
conda activate lisa2

cat /n/stat115/2021/HW3/up_regulated_genes_in_prostate_cancer.txt | cut -f 1 | tail -n +2 > upregulated

lisa oneshot hg38 upregulated_gene_names.txt -b 300 --seed=2556 --save_metadata >  q20.out
```

```
knitr::include_graphics("q20.png")
```

	Rank	sample_id	factor	cell_line	cell_type	tissue	DNase_p_value	ChIP.seq_p_value	H3K27ac_p_value	summary_p_value
1	1	56122	AR	None	None	Prostate	4.888017e-11	1.310725e-15	4.603944e-09	4.107825e-15
2	2	67742	AR	LNCaP	Epithelium	Prostate	1.224405e-10	2.382433e-14	1.214710e-11	7.133183e-14
3	3	56390	AR	None	None	Prostate	4.222683e-13	3.298796e-14	3.614952e-13	8.465451e-14
4	4	76156	AR	LNCaP	Epithelium	Prostate	2.756460e-12	5.403943e-14	7.369021e-13	1.482703e-13
5	5	56206	AR	LNCaP	Epithelium	Prostate	3.148185e-13	9.692336e-14	2.988677e-13	1.781353e-13
6	6	67745	AR	LNCaP	Epithelium	Prostate	6.739982e-12	1.008153e-13	1.070054e-11	2.952083e-13
7	7	33150	GATA3	MCF-7	Epithelium	Breast	1.432497e-10	1.005702e-13	9.187396e-11	3.012590e-13
8	8	50049	AR	None	Epithelium	Prostate	1.634245e-13	1.245871e-12	4.823889e-13	3.335110e-13
9	9	69288	AR	VCaP	Epithelium	Prostate	8.005173e-13	2.445992e-13	1.199729e-12	4.860556e-13
10	10	69275	AR	VCaP	Epithelium	Prostate	2.952911e-12	2.322434e-13	5.558431e-12	6.218914e-13
11	11	69289	AR	VCaP	Epithelium	Prostate	2.169578e-12	2.504426e-13	2.738171e-12	6.226131e-13
12	12	76154	AR	LNCaP	Epithelium	Prostate	5.020725e-12	3.262115e-13	4.878268e-12	8.647527e-13
13	13	56265	AR	CWR22Pc	Epithelium	Prostate	3.128967e-08	4.501136e-13	3.321790e-08	1.350475e-12
14	14	56121	AR	None	None	Prostate	3.079481e-12	6.451863e-13	8.810298e-12	1.509015e-12
15	15	56118	AR	None	None	Prostate	7.630567e-12	9.864627e-13	1.125302e-11	2.431833e-12
16	16	56392	AR	None	None	Prostate	3.140659e-12	1.802802e-12	2.997721e-12	2.486122e-12
17	17	56107	AR	None	None	Prostate	7.139099e-11	8.806276e-13	3.571441e-11	2.547740e-12
18	18	67744	FOXA1	LNCaP	Epithelium	Prostate	2.270180e-08	8.868185e-13	5.704779e-09	2.660150e-12
19	19	8653	AR	VCaP	Epithelium	Prostate	1.049004e-10	1.280926e-12	3.755210e-11	3.672673e-12
20	20	51758	AR	LNCaP	Epithelium	Prostate	1.241166e-11	2.733554e-12	5.876520e-12	4.865663e-12

```
lisa_out <- read.csv('q20.out', sep = '\t')
lisa_out %>% filter(tissue == 'Prostate') %>%
  select(factor, summary_p_value) %>%
  group_by(factor) %>%
  summarize_all(min) %>%
  arrange(summary_p_value) %>%
  head(10)
```

```
## # A tibble: 10 x 2
##    factor summary_p_value
##    <chr>            <dbl>
##  1 AR            4.11e-15
##  2 FOXA1         2.66e-12
##  3 GATA2         9.76e-10
##  4 HOXB13        3.16e- 9
##  5 CTRL          6.02e- 9
##  6 PIAS1         6.23e- 9
##  7 SUMO2         2.16e- 8
##  8 NR3C1         9.51e- 8
##  9 NKX3-1        1.95e- 7
## 10 HDAC3         9.91e- 7
```

These are the top 10 most associated unique transcription factors. We see a lot of overlap with answers to previous questions, especially AR, FOXA1, and HOXB13.

Summary: With HW3, we hope you can see the value of TF motif, TF ChIP-seq, and chromatin accessibility profiles in understanding gene regulation or differential gene expression. We also hope you to appreciate the value of using publicly available data and integration resources in Cistrome to inform your own research explorations.

Part VII: Bonus question

In this course, we could not provide the full answers to HW questions after grading. This is because it is extremely time consuming to find public data which allow students to use all public tools / programs to touch on all the knowledge points in a module and get all the desirable results. As a result, we could only update some of the questions every year, and reuse many questions / data from before.

This year, we are asking students to help us make HW questions for next year. These are completely optional. You will be given a maximum of 5 points, if your contribution is selected for future HW. Since this is a bonus question, we ask students to demonstrate your own initiative and independence, so unfortunately the TAs won't be able to help much.

Bonus Q1:

For batch effect removal, we used to have a perfect example from expression microarrays. With and without batch effect, the clustering, differential expression, and GO analysis give completely different results, also batch effect removal greatly improved the results. Unfortunately with microarray topic removed from the class, we haven't been able to find a good (and simple) RNA-seq example for Part I of HW2. In 2020, the batch effect was very complicated, so students had too much trouble. This year, after testing many public datasets without success, we finally decided to simulate the RNA-seq data used in HW2 Part 1 by artificially adding batch effect to half of the samples. You might have noticed that with or without batch effect removal, even though PCA and clustering look different, the GO analysis give quite similar results. Therefore, we are asking whether you could find a better dataset for HW2 Part I, to show case batch effect removal, differential

expression, clustering (H-cluster, K-means, and PCA), GO, and GESA. We hope you could provide the data, code, as well as the answers to all the questions in HW2 Part I.

Bonus Q2:

There are different machine learning packages available in the public. Caret is an R version which you used in Part II of HW2. Sklearn is a python package for machine learning. For this bonus question, we ask you to rewrite the Part II of HW2, with your Sklearn solution. You might also need pandas and numpy python packages, and some R plotting functions.

I answered this question! I've submitted a Jupyter notebook (BonusQ2.ipynb) as well as an HTML file (BonusQ2.html) with all code and output.