Short Answer Questions

Ans 1) RNA-seq Analysis Workflow

RNA-seq analysis involves several steps:

1. Data Preprocessing: The first step is to assess the quality of the raw sequencing reads using tools like FastQC.
2. Alignment/Mapping: The cleaned reads are mapped to a reference genome using alignment tools such as STAR
3. Quantification: The next step is to count how many reads map to each gene using tools like featureCounts, creating a gene expression matrix.
4. Normalization: RNA-seq data is normalized to adjust for sequencing depth and gene length differences. Tools like DESeq2
5. Differential Expression Analysis: Statistical tools like DESeq2 or EdgeR are used to identify genes that are significantly different between the conditions being studied.

2) Molecular Events Identified with RNA-seq

RNA-seq data can reveal a variety of important molecular events:

1. Gene Expression: Identifies genes that are differentially expressed between different conditions (e.g., disease vs. control) using tools like DESeq2 or EdgeR
2. Alternative Splicing: RNA-seq enables the detection of alternative splicing events (e.g., exon skipping or alternative promoters).
3. **Non-coding RNAs**: RNA-seq identifies non-coding RNAs like microRNAs and long non-coding RNAs (lncRNAs), which regulate gene expression.
4. **Transcript Isoforms**: RNA-seq detects different isoforms generated by alternative splicing, alternative polyadenylation, or alternative promoters

3) Single-cell RNA-seq vs. Bulk RNA-seq

Single-cell RNA-seq (scRNA-seq)

Pros:

- Cell-to-Cell Variability: scRNA-seq provides insights into gene expression at the individual cell level, capturing cellular diversity within a sample.
- Rare Cell Types: It can identify rare or previously undetected cell types or states within a population.

Cons:

- High Cost: Single-cell RNA-seq is more expensive and technically challenging due to the need for isolating and amplifying single cells.
- Lower Sensitivity: There’s higher technical noise and lower sensitivity in detecting lowly expressed genes at the single-cell levell

Bulk RNA-seq

Pros:

- Cost-effective: Bulk RNA-seq is more affordable than scRNA-seq, especially for large sample sizes.
- Higher Sensitivity: It is more sensitive to detecting gene expression in larger populations of cells.

Cons:

- Averaged Data: Bulk RNA-seq averages gene expression across all cells, meaning it cannot detect cell-specific expression patterns or rare cell types.
- No Cell Heterogeneity: Cell-to-cell variation is lost in bulk analysis, which can be a limitation in heterogeneous tissues or tumors.

Ans 2) Epigenetic events are chemical changes that alter gene expression without changing the DNA sequence. These events play a crucial role in regulating gene activity and are involved in development, disease, and response to environmental factors. The main epigenetic events in the human genome include:

1. DNA Methylation: DNA methylation involves the addition of a methyl group to the 5' carbon of cytosine residues, typically in CpG dinucleotides. Methylation can repress gene expression by preventing transcription factor binding or recruiting repressive proteins. Abnormal DNA methylation patterns are associated with diseases like cancer.

   Techniques: Methylation-specific PCR, and Methyl-seq are used to map DNA methylation patterns.

2. Histone Modifications: Histones are proteins around which DNA is wrapped, and their chemical modifications (such as acetylation, methylation, phosphorylation) influence chromatin structure. Acetylation usually promotes gene activation by loosening chromatin, while methylation can either activate or repress genes, depending on the site.

   Techniques: ChIP-seq (Chromatin Immunoprecipitation Sequencing) allows the profiling of specific histone modifications. Mass spectrometry can also be used to detect histone modifications.

3. Chromatin Remodeling: Chromatin remodeling involves the repositioning or restructuring of nucleosomes, which makes the DNA more or less accessible to transcription machinery. This is essential for gene regulation and maintaining chromatin structure during cell division.

   Techniques: DNase-seq are used to study chromatin accessibility and structural changes.

4. Non-coding RNA Regulation: Non-coding RNAs (ncRNAs), including microRNAs and long non-coding RNAs (lncRNAs), regulate gene expression by interacting with DNA, RNA, or proteins. They can silence genes through mechanisms such as RNA interference or chromatin modification.

   Techniques: RNA-seq can be used to profile non-coding RNA expression, while ChIP-seq can help study how they affect chromatin.

Ans 3) How do epigenetic modifications influence the expression of immune-related genes in breast cancer and affect the tumor microenvironment (TME)?

By integrating epigenomic (DNA methylation), transcriptomic (RNA-seq), and clinical annotation (such as patient survival and immune cell infiltration), we can investigate the role of epigenetic changes in modulating immune-related genes, potentially affecting immune response and patient prognosis in breast cancer.

Hypothesis:

Aberrant DNA methylation in the promoter regions of immune-related genes, such as those involved in immune checkpoint regulation (e.g., PD-1, CTLA-4), affects their expression in breast cancer. This alteration contributes to immune evasion, influencing tumor progression and patient survival.

Plan:

1. Data Collection and Preprocessing:
   - Retrieve breast cancer data from TCGA, which includes DNA methylation, RNA-seq, and clinical annotations (survival data, immune cell infiltration).
   - Process RNA-seq data to quantify gene expression levels and DNA methylation data to determine methylation status at promoter regions of immune-related genes (e.g., PD-1, CTLA-4, CD274).
2. Selection of Immune-Related Genes:
   - Focus on immune checkpoint genes (e.g., PD-1, CTLA-4, CD274), cytokine receptors, and other genes that regulate immune responses.
   - Use Gene Ontology (GO) enrichment analysis or ImmPort database to identify relevant immune-related genes in breast cancer.
3. Integration of Epigenomic and Transcriptomic Data:
   - Investigate whether methylation in the promoter regions of these immune-related genes is correlated with expression levels of the corresponding genes.
   - Use correlation analysis (e.g., Pearson or Spearman) to study the relationship between methylation status and gene expression in immune-related genes.
4. Immune Infiltration Analysis:
   - Analyze the correlation between immune-related gene expression and immune cell infiltration (e.g., T-cells, macrophages) in the TME using tools like CIBERSORT or EPIC to estimate immune cell composition from RNA-seq data.
5. Survival Analysis:
   - Perform Kaplan-Meier survival analysis to assess the impact of DNA methylation and immune gene expression on breast cancer patient survival.
   - Use Cox proportional hazards regression to identify significant predictors of survival, incorporating both epigenetic and immune cell data.
6. Statistical Analysis and Multi-Omic Integration:
   - Use multi-omic integration methods such as MOFA or iCluster to combine methylation, expression, and immune cell data and uncover hidden relationships that affect tumor progression and immune evasion.
   - Perform differential expression analysis (using DESeq2 or EdgeR) to identify immune-related genes whose expression is significantly associated with different methylation patterns.

Long Answer Questions

DNA-SEQ Analysis

1) **Steps in Termius :-**

```
#To start an interactive session we need to ask for some space on a worker node
qlogin -pe smp 1 -l h_vmem=4G -l h_rt=1:0:0

### Get the Reference folder
cp -v /data/teaching/bci_teaching/assignment/DNAseq/tumour_R1.fq.gz ./
cp -v /data/teaching/bci_teaching/assignment/DNAseq/tumour_R2.fq.gz ./
cp -v /data/teaching/bci_teaching/assignment/DNAseq/germline_R1.fq.gz ./
cp -v /data/teaching/bci_teaching/assignment/DNAseq/germline_R2.fq.gz ./

#load modules
module load bowtie2
## Run bowtie2-build, the index building part of Bowtie2
bowtie2-build Homo_sapiens.GRCh38.dna.chromosome.17.fa
Bowtie2Idx/GRCh38.108
Module samtools

#Then we need to make a directory to store our alignments in.
mkdir Align
```

```
#Now we're ready to start our alignments.
```

For the gremlin fastq files:

```
time bowtie2 -p 4 \
        --rg ID:germline \
        --rg SM:germline \
        --rg PL:ILLUMINA \
        --rg LB:germline \
        -x Reference/Bowtie2Idx/GRCh38.108.chr17 \
        -1 assignment/DNAseq/germline_R1.fq.gz  \
        -2 assignment/DNAseq/germline_R2.fq.gz  |
        samtools sort -o Align/germline.bam -
```

For the tumor fastq files:

```
time bowtie2 -p 4 \
        --rg ID:tumor \
        --rg SM:tumor \
        --rg PL:ILLUMINA \
        --rg LB:tumor \
        -x Reference/Bowtie2Idx/GRCh38.108.chr17 \
        -1 assignment/DNAseq/tumour_R1.fq.gz\
        -2 assignment/DNAseq/tumour_R2.fq.gz|
        samtools sort -o Align/tumor.bam -
```

## Mark Duplicates

```
module load java
module load gatk
```

2) For the germline bam file, marking duplicates:

```
 gatk --java-options "-Xmx1G" MarkDuplicates \
        -I Align/germline.bam \
        -M QC2/germline.marked \
        -O Align/germline.marked.bam
```

For the tumor bam file, marking duplicates:

```
 gatk --java-options "-Xmx1G" MarkDuplicates \
        -I Align/tumor.bam \
        -M QC2/tumor.marked \
        -O Align/tumor.marked.bam
```

## Base Quality Score Recalibration

For the germline sample,

```
gatk --java-options "-Xmx1G" BaseRecalibrator \
        -I Align/germline.marked.bam \
        -R Reference/Homo_sapiens.GRCh38.108.dna.chromosome.17.fa \
--known-sitesReference/
gatkResources/resources_broad_hg38_v0_1000G_omni2.5.hg38.noCHR.vcf \
-o Align/germline.table
```

For the tumor sample,

```
gatk --java-options "-Xmx1G" BaseRecalibrator \
        -I Align/tumor.marked.bam \
        -R Reference/Homo_sapiens.GRCh38.108.dna.chromosome.17.fa \
--known-sitesReference/
gatkResources/resources_broad_hg38_v0_1000G_omni2.5.hg38.noCHR.vcf \
-o Align/tumor.table
```

#Applying the model,
For the germline sample,

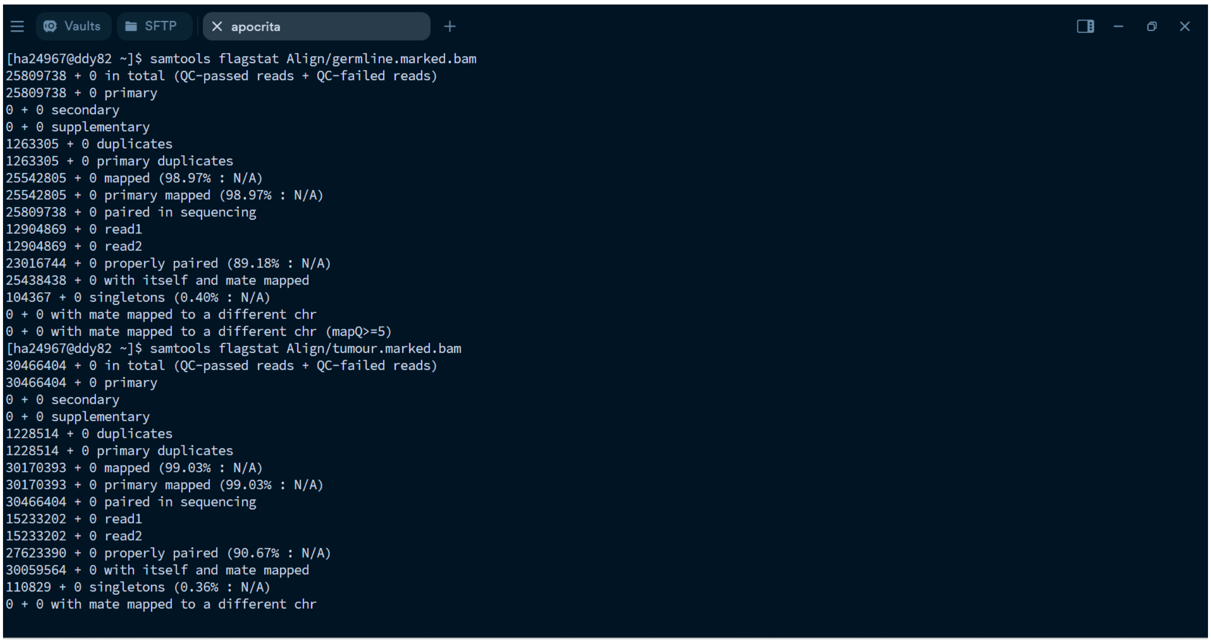
```
gatk --java-options "-Xmx1G" ApplyBQSR \
        -R Reference/Homo_sapiens.GRCh38.108.dna.chromosome.17.fa \
        -I Align/germline.marked.bam \
        --bqsr-recal-file Aligno/germline.table \
        -O Align/germline.recalib.bam
```

For the tumor sample,

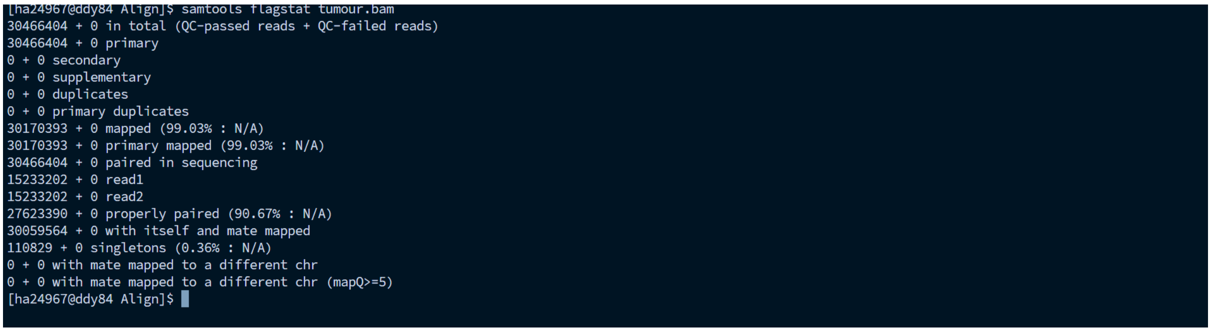
```
gatk --java-options "-Xmx1G" ApplyBQSR \
        -R Reference/Homo_sapiens.GRCh38.108.dna.chromosome.17.fa \
        -I Align/tumor.marked.bam \
        --bqsr-recal-file Align/tumor.table \
        -O Align/tumor.recalib.bam
```

samtools flagstat

```
samtools flagstat germline.bam

samtools flagstat tumour.bam
```



```
[ha24967@ddy82 ~]$ samtools flagstat Align/germline.marked.bam
25809738 + 0 in total (QC-passed reads + QC-failed reads)
25809738 + 0 primary
0 + 0 secondary
0 + 0 supplementary
1263305 + 0 duplicates
1263305 + 0 primary duplicates
25542805 + 0 mapped (98.97% : N/A)
25542805 + 0 primary mapped (98.97% : N/A)
25809738 + 0 paired in sequencing
12904869 + 0 read1
12904869 + 0 read2
23016744 + 0 properly paired (89.18% : N/A)
25438438 + 0 with itself and mate mapped
104367 + 0 singletons (0.40% : N/A)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
[ha24967@ddy82 ~]$ samtools flagstat Align/tumour.marked.bam
30466404 + 0 in total (QC-passed reads + QC-failed reads)
30466404 + 0 primary
0 + 0 secondary
0 + 0 supplementary
1228514 + 0 duplicates
1228514 + 0 primary duplicates
30170393 + 0 mapped (99.03% : N/A)
30170393 + 0 primary mapped (99.03% : N/A)
30466404 + 0 paired in sequencing
15233202 + 0 read1
15233202 + 0 read2
27623390 + 0 properly paired (90.67% : N/A)
30059564 + 0 with itself and mate mapped
110829 + 0 singletons (0.36% : N/A)
0 + 0 with mate mapped to a different chr
```



```
[ha24967@ddy84 Align]$ samtools flagstat tumour.bam
30466404 + 0 in total (QC-passed reads + QC-failed reads)
30466404 + 0 primary
0 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
0 + 0 primary duplicates
30170393 + 0 mapped (99.03% : N/A)
30170393 + 0 primary mapped (99.03% : N/A)
30466404 + 0 paired in sequencing
15233202 + 0 read1
15233202 + 0 read2
27623390 + 0 properly paired (90.67% : N/A)
30059564 + 0 with itself and mate mapped
110829 + 0 singletons (0.36% : N/A)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
[ha24967@ddy84 Align]$
```

4) For VarScan

```
# Varscan2
module load java
module load samtools

# Make a VCF directory
mkdir VCF
samtools mpileup \
    -q 20 \
    -f Reference/Homo_sapiens.GRCh38.108.dna.chromosome.17.fa \
    Align/tumor.recal.bam |
java -jar VarScan.v2.4.3.jar mpileup2snp \
    --min-coverage 20 \
    --min-avg-qual 20 \
    --min-read2 4 \
    --p-value 0.2 \
    --min-var-freq 0.01 \
    --strand-filter 1 \
    --output-vcf 1 > vcf/tumor.vcf
```

5) Annotation

module load annovar

First we convert to annovar

```
convert2annovar.pl --format vcf4 \
        vcf/tumor.vcf \
        --includeinfo \
        --filter PASS \
        --outfile vcf/tumor.pass.vcf
```

For 1000G:

```
annotate_variation.pl -filter \
        -dbtype 1000g2015aug_all \
        -buildver hg38 \
        -out vcf/tumor \
        vcf/tumor.pass.vcf \
        Reference/humandb/ \
        -maf 0.01
```

For exome sequencing project:

```
annotate_variation.pl -filter \
        -dbtype esp6500siv2_all \
        -buildver hg38 \
        -out vcf/tumoresp \
      vcf/tumorsample.hg38_ALL.sites.2015_08_filtered \
        Reference/humandb/ \
        -score_threshold 0.01
```

To annotate with gene names, dbSNP id and cosmic id:

For 1000G:

```
table_annovar.pl \
        -buildver hg38 \

        -out vcf/tumors \
      vcf/tumorsample.hg38_ALL.sites.2015_08_filtered\
        Reference/humandb/ \
        -remove \
        -otherinfo \
        -protocol refgene,avsnp150,cosmic92_coding,cytoband \
        -operation g,f,f,r -nastring .
```

To filter out the variants with a variant allele frequency of <10%

```
require(tidyverse)
variants <- read.delim("/Users/dataramvenkatasrikarkeshav/tumors.hg38_multianno.txt", header =
FALSE)
variants
headings <- c("chr", "position","id", "ref", "alt", "qual", "filter", "info", "format" ,"sample")
Annotated_variants <- setNames(variants[-1,], c(variants[1,1:13] %>% unlist(), headings))
Annotated_variants
```

```
headings <- str_split(Annotated_variants_new$format[1], ":") %>% unlist()
AlleleCounts <- str_split(Annotated_variants_new$sample, ":") %>% do.call("rbind", .) %>%
as.data.frame() %>% setNames(headings)
AlleleCounts <- mutate(AlleleCounts, FREQ = gsub("%", "", FREQ) %>% as.numeric())
AlleleCounts
Annotated_variants <- cbind(Annotated_variants, AlleleCounts)
Annotated_variants
separate(Annotated_variants, sample, into = headings, sep = “:")
```

```
Annotated_variants_exonic <- subset(Annotated_variants,Func.refgene == "exonic" &
ExonicFunc.refgene != "synonymous SNV”)

new_variants <- subset(Annotated_variants_exonic, FREQ > 10)
```

RNA Analysis

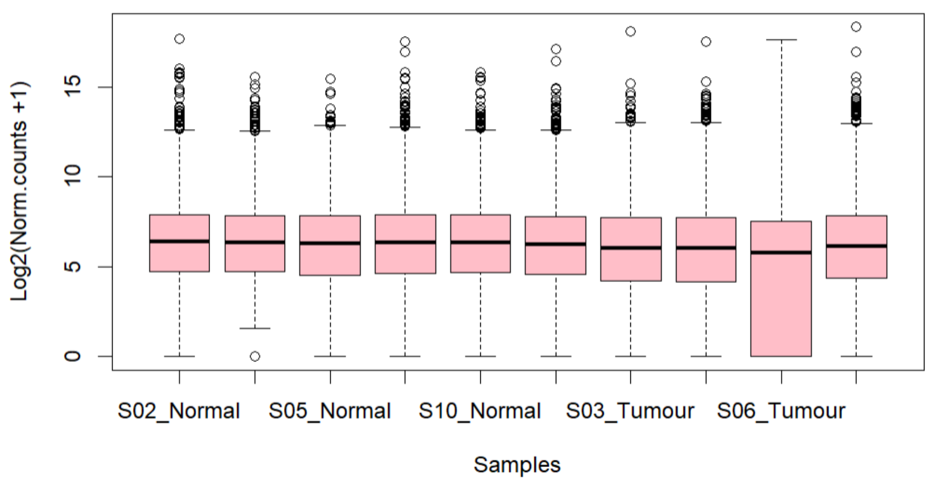
1)

```
  # load the library
  library(DESeq2)

counts_1 <- read.delim("C:/Users/dell/Downloads/Data_2/row_count_data.nodup.txt", header =
T,row.names = 1)
counts_1 <- ceiling(counts_1)
samples_1 <- read.delim("C:/Users/dell/Downloads/Data_2/sample_groups.txt", header =
TRUE, row.names = 1, stringsAsFactors = TRUE)

head(counts_1)
head(samples_1)

# Create the DEseq2DataSet object
dds <- DESeqDataSetFromMatrix(countData = counts_1,
                              colData = samples_1,
                              design = ~ Patient + Group)

dds
dim(dds)
```

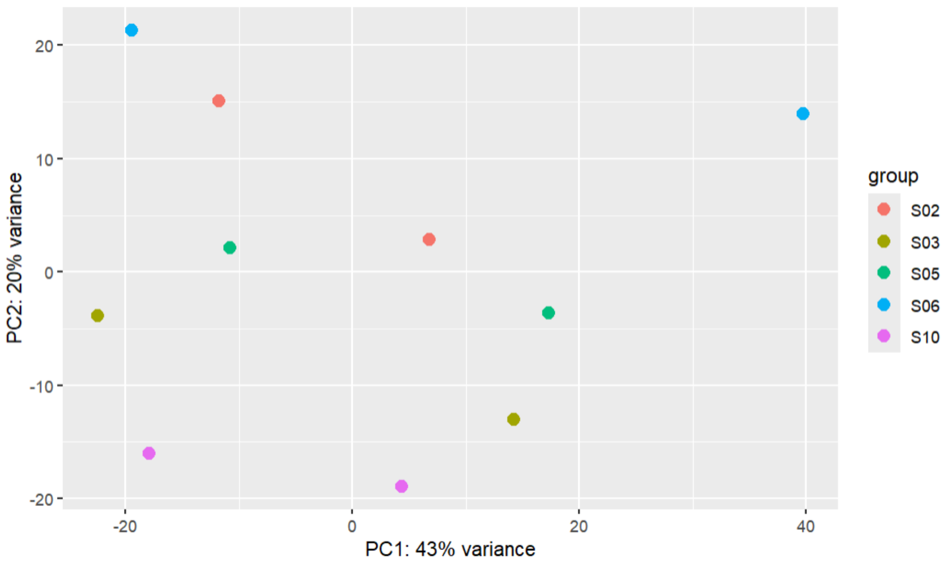
```
#a popular filter is to ensure at least X samples with a count of 10 or more, where X can be chosen as the sample size of the smallest group of samples
keep <- rowSums(counts(dds) >= 10) >= 3
dds <- dds[keep,]

dds

dds <- DESeq(dds)
```

2) Boxplot -
3)

```
# Extracting normalised counts and vsdtransformed counts
norm.counts <- counts(dds, normalized=TRUE)
write.csv(norm.counts, file = "normal_counts.csv", row.names = TRUE)
# Boxplot for normalized counts
boxplot(log2(norm.counts + 1),
     col = "pink",
     main = "Boxplot of Normalized Gene Expression",
     ylab = "Log2(Norm.counts +1)",
     xlab = "Samples",
)
#vst
vsd <- vst(dds, blind = FALSE)

# Inspect vst data
head(assay(vsd),3)
```

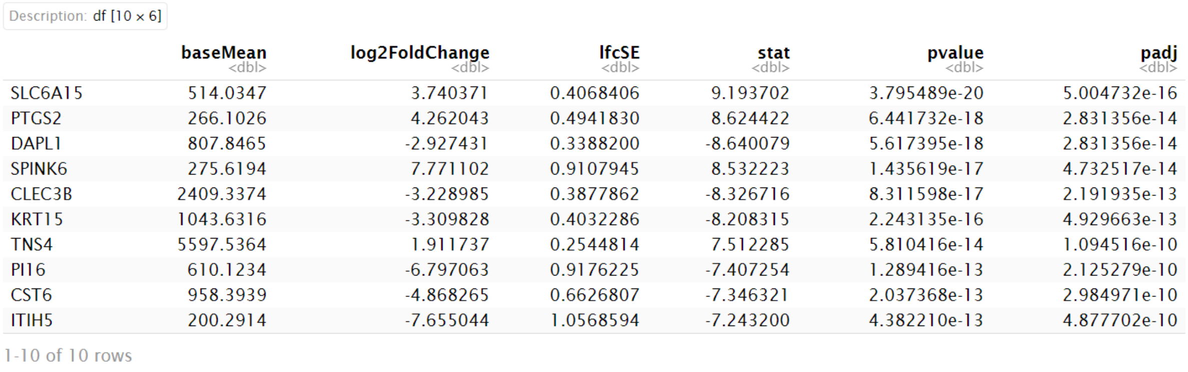


```
#PCA Plot
plotPCA(vsd, intgroup = "Patient")
```



3)

```
res <- results(dds, contrast = c("Group", "Tumour", "Normal"))
# View summary of results
summary(res)
head(res)
#Exporting results to csv
write.csv(as.data.frame(res), file = "result.csv", row.names = TRUE)
# Sort results by adjusted p-value
res <- res[order(res$padj), ]

# Extract top 10 differentially expressed genes
top10 <- as.data.frame(res)[1:10, ]

# Save the top 10 DE genes to a CSV file
write.csv(top10, file = "Top10_genes.csv", row.names = TRUE)

# View top 10 genes
top10

top10_genes <- rownames(top10)
```



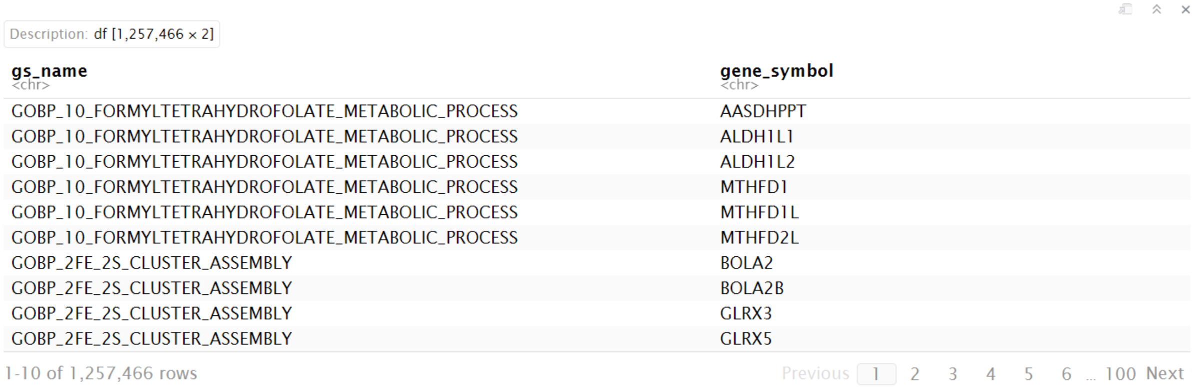
Description: df [10 x 6]

	baseMean <dbl>	log2FoldChange <dbl>	lfcSE <dbl>	stat <dbl>	pvalue <dbl>	padj <dbl>
SLC6A15	514.0347	3.740371	0.4068406	9.193702	3.795489e-20	5.004732e-16
PTGS2	266.1026	4.262043	0.4941830	8.624422	6.441732e-18	2.831356e-14
DAPL1	807.8465	-2.927431	0.3388200	-8.640079	5.617395e-18	2.831356e-14
SPINK6	275.6194	7.771102	0.9107945	8.532223	1.435619e-17	4.732517e-14
CLEC3B	2409.3374	-3.228985	0.3877862	-8.326716	8.311598e-17	2.191935e-13
KRT15	1043.6316	-3.309828	0.4032286	-8.208315	2.243135e-16	4.929663e-13
TNS4	5597.5364	1.911737	0.2544814	7.512285	5.810416e-14	1.094516e-10
PI16	610.1234	-6.797063	0.9176225	-7.407254	1.289416e-13	2.125279e-10
CST6	958.3939	-4.868265	0.6626807	-7.346321	2.037368e-13	2.984971e-10
ITIH5	200.2914	-7.655044	1.0568594	-7.243200	4.382210e-13	4.877702e-10

1-10 of 10 rows

4)

```
BiocManager::install("clusterProfiler")
require(clusterProfiler)

install.packages("msigdbr")
require(msigdbr)

#we also need R package dplyr for data wrangling
require(dplyr)

go_gene_sets <- msigdbr(species = "human", category = "C5")  #Load C5

msigdbr_t2g <- go_gene_sets %>% dplyr::distinct(gs_name,gene_symbol) %>% as.data.frame()# make a dataframe of the genesets and the corresponding Ensemble gene ids
msigdbr_t2g
```



Description: df [1,257,466 × 2]

gs_name <chr>	gene_symbol <chr>
GOBP_10_FORMYLTETRAHYDROFOLATE_METABOLIC_PROCESS	AASDHPPT
GOBP_10_FORMYLTETRAHYDROFOLATE_METABOLIC_PROCESS	ALDH1L1
GOBP_10_FORMYLTETRAHYDROFOLATE_METABOLIC_PROCESS	ALDH1L2
GOBP_10_FORMYLTETRAHYDROFOLATE_METABOLIC_PROCESS	MTHFD1
GOBP_10_FORMYLTETRAHYDROFOLATE_METABOLIC_PROCESS	MTHFD1L
GOBP_10_FORMYLTETRAHYDROFOLATE_METABOLIC_PROCESS	MTHFD2L
GOBP_2FE_2S_CLUSTER_ASSEMBLY	BOLA2
GOBP_2FE_2S_CLUSTER_ASSEMBLY	BOLA2B
GOBP_2FE_2S_CLUSTER_ASSEMBLY	GLRX3
GOBP_2FE_2S_CLUSTER_ASSEMBLY	GLRX5

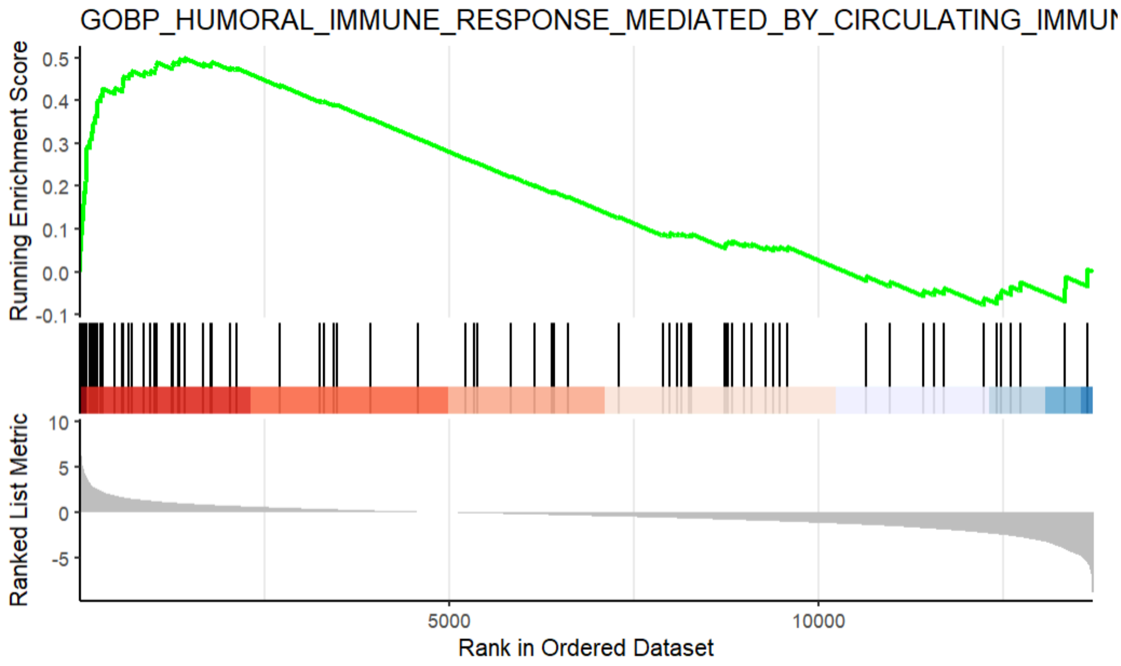
1-10 of 1,257,466 rows

```
DE.res <- read.csv("C:/Users/dell/Downloads/result.csv", header = TRUE, row.names = 1)
DE.res.sign <- subset(DE.res, padj < 0.05 & log2FoldChange > 1)
DE.res.sign

BiocManager::install("fgsea")
library(fgsea)

install.packages("data.table")
library(data.table)

BiocManager::install("qusage")
library(qusage)

install.packages("ggplot2")
library(ggplot2)

library(clusterProfiler)
gmt_1 <- read.gmt("C:/Users/dell/Downloads/Data_2/c5.go.bp.v7.4.symbols.gmt")

DE.res.ranked <- DE.res[order(DE.res$log2FoldChange, decreasing = T), ]
DE.ranks <- setNames(DE_res_ranked$log2FoldChange, DE_res_ranked$gene.symbol)

ranked_genes <- res$log2FoldChange  # Use log2FoldChange from DESeq2 results
```

```
names(ranked_genes) <- rownames(res)  # Associate gene names with values
ranked_genes <- ranked_genes[order(ranked_genes, decreasing = TRUE)]  # Sort

gsea_res <- GSEA(
  geneList = ranked_genes,
  TERM2GENE = gmt_1,
  pvalueCutoff = 1
)

gsea_table <- as.data.frame(gsea_res)

top_10_pathways <- gsea_table[order(gsea_table$qvalue), ][1:10, ]
print(top_10_pathways)
```

5)

```
BiocManager::install("enrichplot")
library(enrichplot)
gsea_table <- as.data.frame(gsea_res)
## Perform enrichment analysis and write results in a .csv file
upregulated <- gsea_table[gsea_table$NES > 0, ]
downregulated <- gsea_table[gsea_table$NES < 0, ]

# Extract top 2 from each category based on FDR q-values
top_upregulated <- upregulated[order(upregulated$qvalue), ][1:2, ]
top_downregulated <- downregulated[order(downregulated$qvalue), ][1:2, ]

# Enrichment plot for the first upregulated gene set
gseaplot2(gsea_res, geneSetID = top_upregulated$ID[1],
          title = top_upregulated$Description[1])

# Enrichment plot for the second upregulated gene set
gseaplot2(gsea_res, geneSetID = top_upregulated$ID[2],
          title = top_upregulated$Description[2])
```



CHIP-SEQ Analysis

Steps:-

```
qlogin -pe smp 5 -l h_vmem=8G -l h_rt=6:0:0
```

Copy two fastq files from shared resource directory “/data/teaching/bci_teaching/assignment/ChIPseq to ‘cp -vR /data/teaching/bci_teaching/assignment/CHIP’

Once you are assigned a node, cd into your working directory

```
$ cd ~/CHIP/
```

```
mkdir CHIP
cd CHIP
```

```
cp /data/teaching/bci_teaching/ChIP_seq/CAPAN1_H3K4me1_chr12.fq ./
cp /data/teaching/bci_teaching/ChIP_seq/CAPAN1_H3K4me3_chr12.fq ./
cp /data/teaching/bci_teaching/ChIP_seq/CAPAN1_input_chr12.rmdup.bam ./
```

Then we use bowtie2 to align it to chr12

```
module load bowtie2
bowtie2 -p 4 -q --local -x /data/teaching/bci_teaching/reference_data/chr12_hg38_bowtie2
-U
CAPAN1_H3K4me1_chr12.fq -S CAPAN1_H3K4me1_chr12.sam
```

```
bowtie2 -p 4 -q --local -x
/data/teaching/bci_teaching/reference_data/chr12_hg38_bowtie2 -U
CAPAN1_H3K4me3_chr12.fq -S CAPAN1_H3K4me3_chr12.sam
```

```
head CAPAN1_H3K4me1_chr12.sam
head CAPAN1_H3K4me3_chr12.sam
```

```
module load samtools
```

First, convert SAM to BAM format

```
$ samtools view -S -b CAPAN1_H3K4me1_chr12.sam > CAPAN1_H3K4me1_chr12.bam
$ samtools view -S -b CAPAN1_H3K4me3_chr12.sam > CAPAN1_H3K4me3_chr12.bam
```

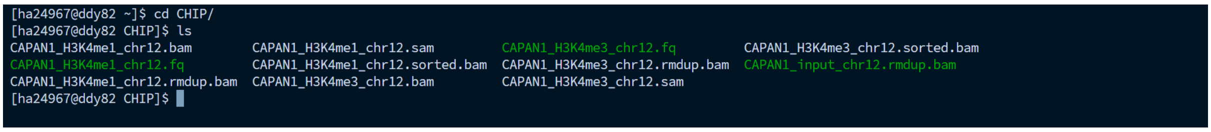
Next, sort these alignments with regard to their position in the reference genome

```
$ samtools sort CAPAN1_H3K4me1_chr12.bam -o CAPAN1_H3K4me1_chr12.sorted.bam
$ samtools sort CAPAN1_H3K4me3_chr12.bam -o CAPAN1_H3K4me3_chr12.sorted.bam
```

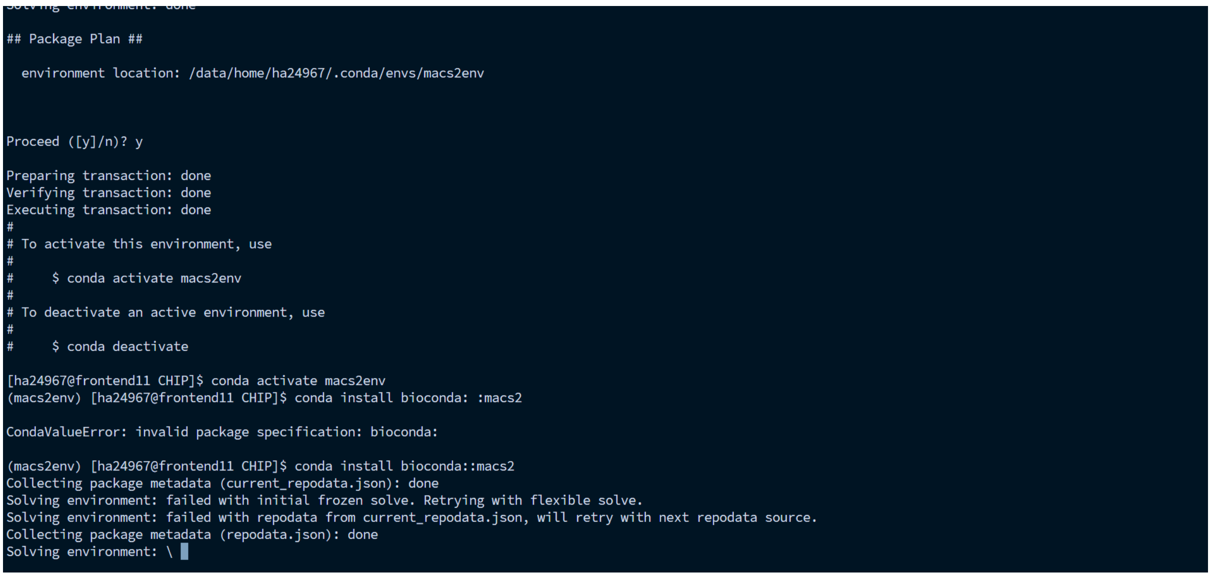
Remove PCR duplicates

```
$ samtools rmdup -s CAPAN1_H3K4me1_chr12.sorted.bam
CAPAN1_H3K4me1_chr12.rmdup.bam
$ samtools rmdup -s CAPAN1_H3K4me3_chr12.sorted.bam
CAPAN1_H3K4me3_chr12.rmdup.bam
```

“ Macs2 Not working, Technical issues “



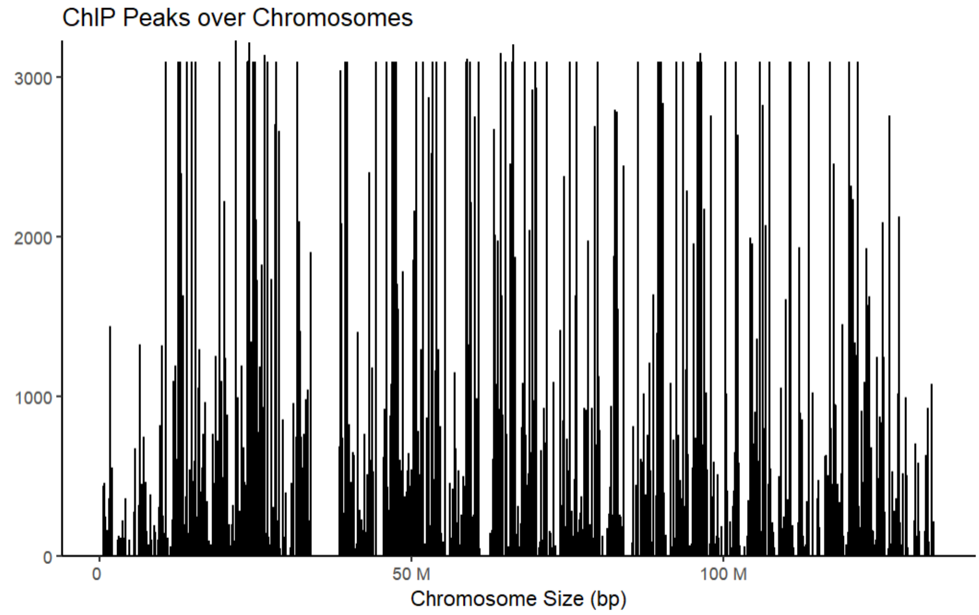
```
[ha24967@ddy82 ~]$ cd CHIP/
[ha24967@ddy82 CHIP]$ ls
CAPAN1_H3K4me1_chr12.bam        CAPAN1_H3K4me1_chr12.sam         CAPAN1_H3K4me3_chr12.fq         CAPAN1_H3K4me3_chr12.sorted.bam
CAPAN1_H3K4me1_chr12.fq         CAPAN1_H3K4me1_chr12.sorted.bam  CAPAN1_H3K4me3_chr12.rmdup.bam  CAPAN1_input_chr12.rmdup.bam
CAPAN1_H3K4me1_chr12.rmdup.bam  CAPAN1_H3K4me3_chr12.bam         CAPAN1_H3K4me3_chr12.sam
[ha24967@ddy82 CHIP]$
```



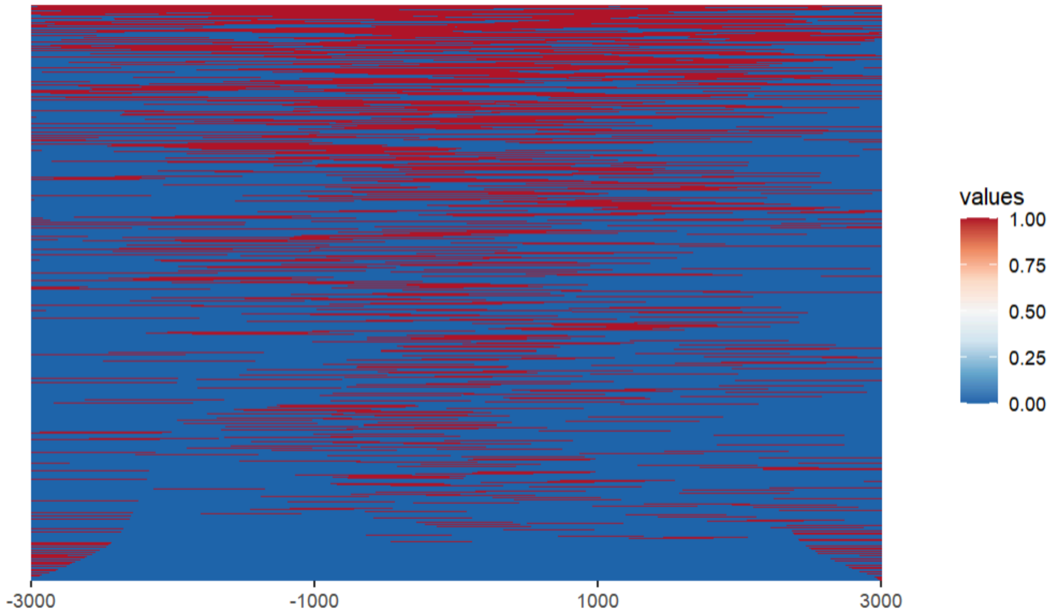
```
Solving environment: done

## Package Plan ##

  environment location: /data/home/ha24967/.conda/envs/macs2env



Proceed ([y]/n)? y

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate macs2env
#
# To deactivate an active environment, use
#
#     $ conda deactivate

[ha24967@frontend11 CHIP]$ conda activate macs2env
(macs2env) [ha24967@frontend11 CHIP]$ conda install bioconda: :macs2

CondaValueError: invalid package specification: bioconda:

(macs2env) [ha24967@frontend11 CHIP]$ conda install bioconda::macs2
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Solving environment: failed with repodata from current_repodata.json, will retry with next repodata source.
Collecting package metadata (repodata.json): done
Solving environment: \
```

ChIPseeker

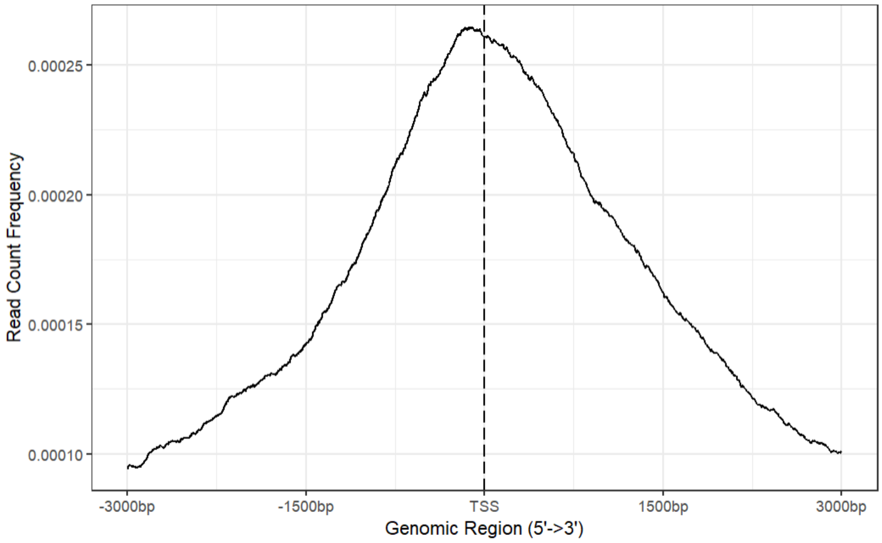
```
BiocManager::install("ChIPseeker")
if (!require("BiocManager", quietly = TRUE))
    install.packages("BiocManager")
BiocManager::install("TxDb.Hsapiens.UCSC.hg19.knownGene")

library(ChIPseeker)
library(TxDb.Hsapiens.UCSC.hg19.knownGene)

setwd("C:/Users/dell/Downloads/Data_2")
getwd()
files <- list("GSM1919984_CFPAC1.crKLF5.ELF3_vs_INPUT.CAPAN1_peaks.bed",
              "GSM1919986_CFPAC1.wtKLF5.ELF3_vs_INPUT.CAPAN1_peaks.bed",
              "GSM1919985_CFPAC1.crKLF5_FOXA1_vs_INPUT.CAPAN1_peaks.bed",
              "GSM1919987_CFPAC1.wtKLF5_FOXA1_vs_INPUT.CAPAN1_peaks.bed")
peak <- readPeakFile(files[[3]], header=F)
Peaks <- lapply(files, readPeakFile)
covplot(peak, weightCol="V5", chrs="chr12")
```



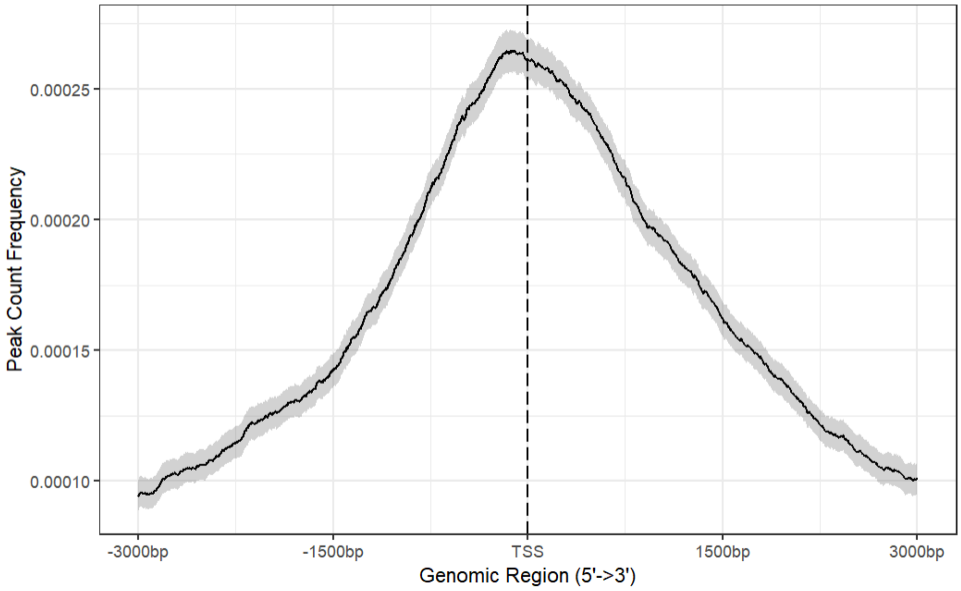
```
txdb19 <- TxDb.Hsapiens.UCSC.hg19.knownGene
library(clusterProfiler)
promoter <- getPromoters(TxDb=txdb19, upstream=3000, downstream=3000)
tagMatrix <- getTagMatrix(peak, windows=promoter)
tagHeatmap(tagMatrix)
```



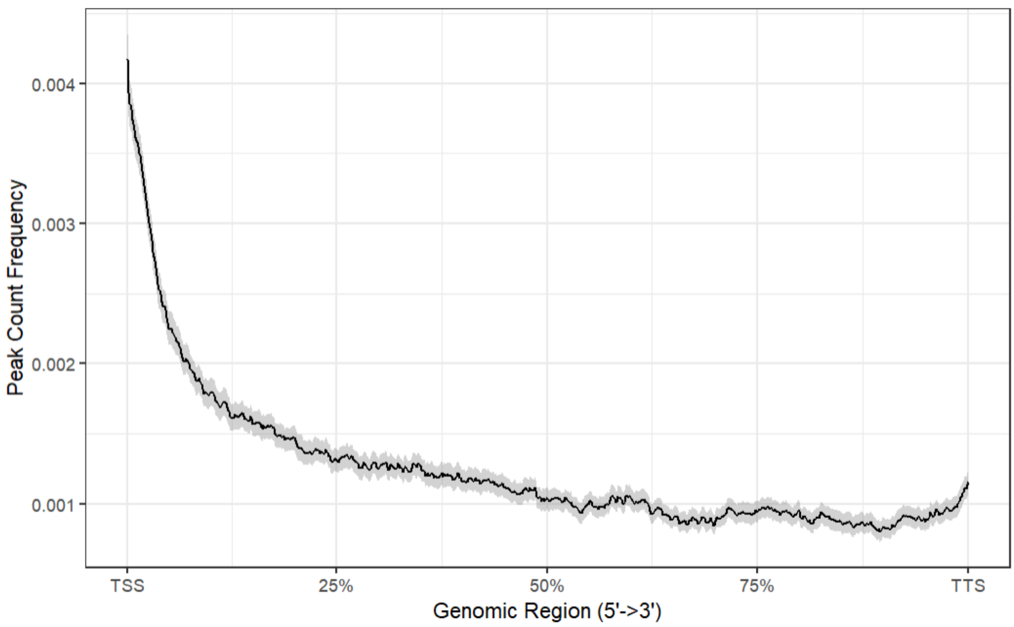
```
plotAvgProf(tagMatrix, xlim=c(-3000, 3000),
 xlab="Genomic Region (5'->3')", ylab = "Read Count Frequency")
```



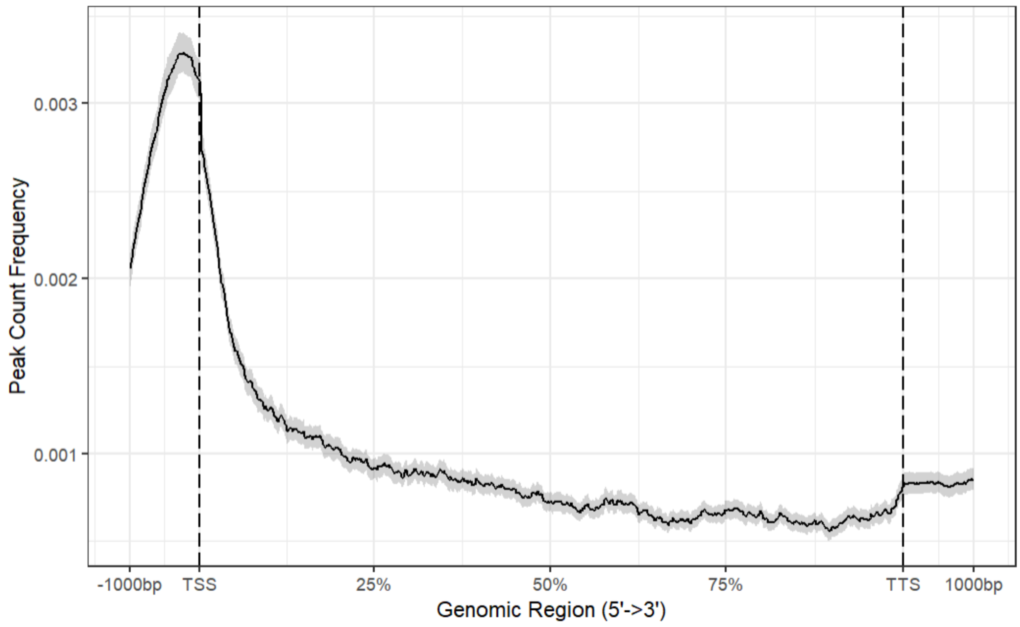
```
plotAvgProf(tagMatrix, xlim=c(-3000, 3000), conf = 0.95, resample = 1000)
```



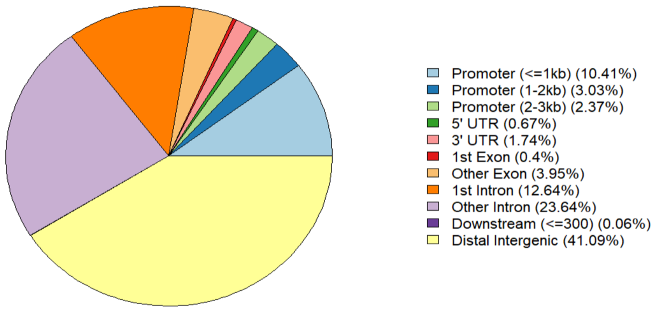
```
genebody <- getBioRegion(TxDb = txdb19, by = "gene", type = "body")
matrix_no_flankextension <- getTagMatrix(peak,windows = genebody, nbin = 800)
plotPeakProf(matrix_no_flankextension,conf = 0.95)
```



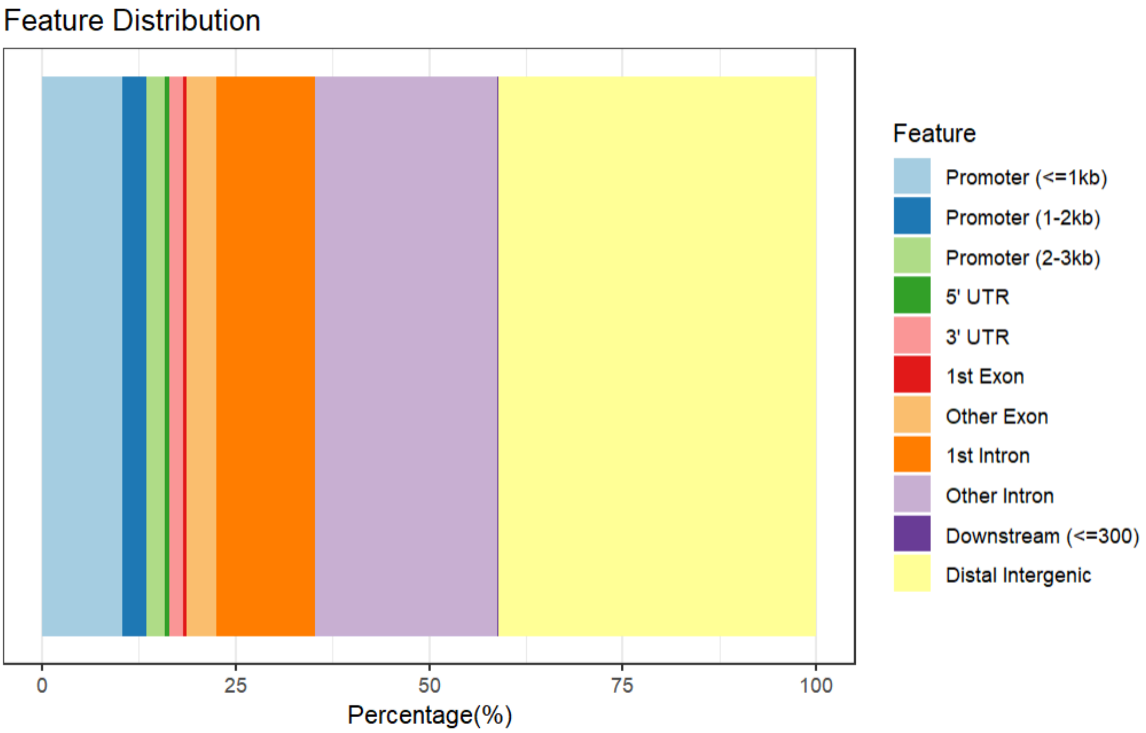
```
matrix_actual_extension <- getTagMatrix(peak,windows = genebody, nbin = 800,
upstream = 1000,downstream = 1000)
plotPeakProf(matrix_actual_extension,conf = 0.95)
```



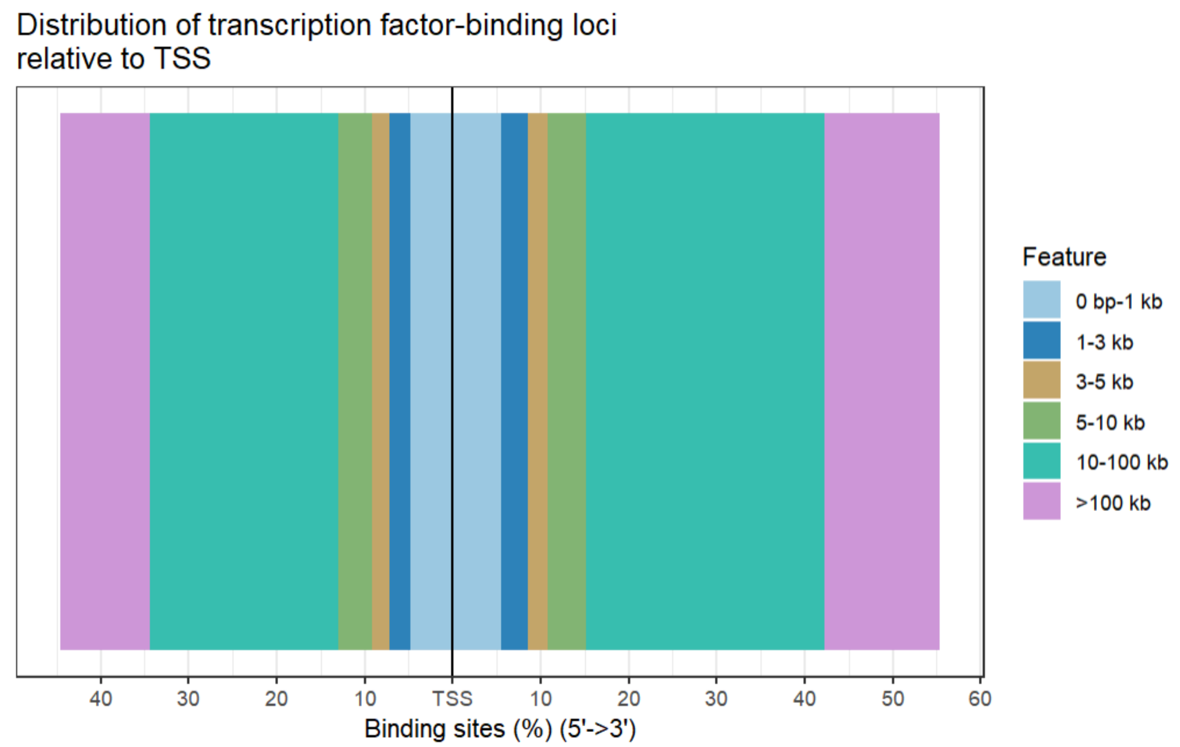
```
if (!require("BiocManager", quietly = TRUE))
 install.packages("BiocManager")
BiocManager::install("org.Hs.eg.db")
library(org.Hs.eg.db)
peakAnno <- annotatePeak(peak, tssRegion=c(-3000, 3000),
                    TxDb=txdb19, annoDb="org.Hs.eg.db")
plotAnnoPie(peakAnno)
```



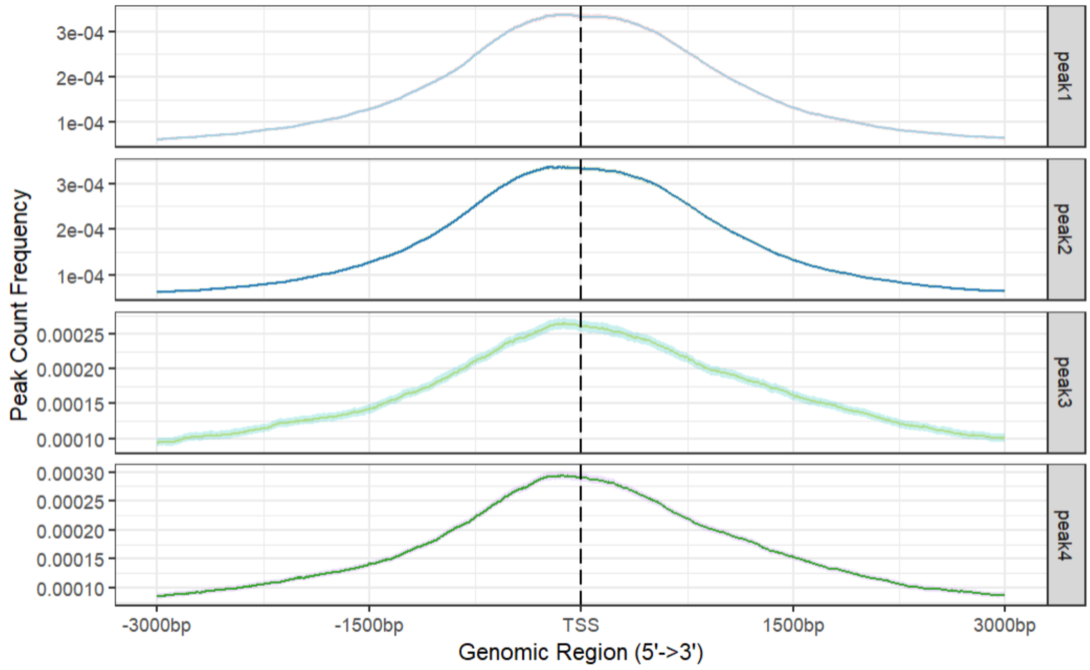
```
plotAnnoBar(peakAnno)
```



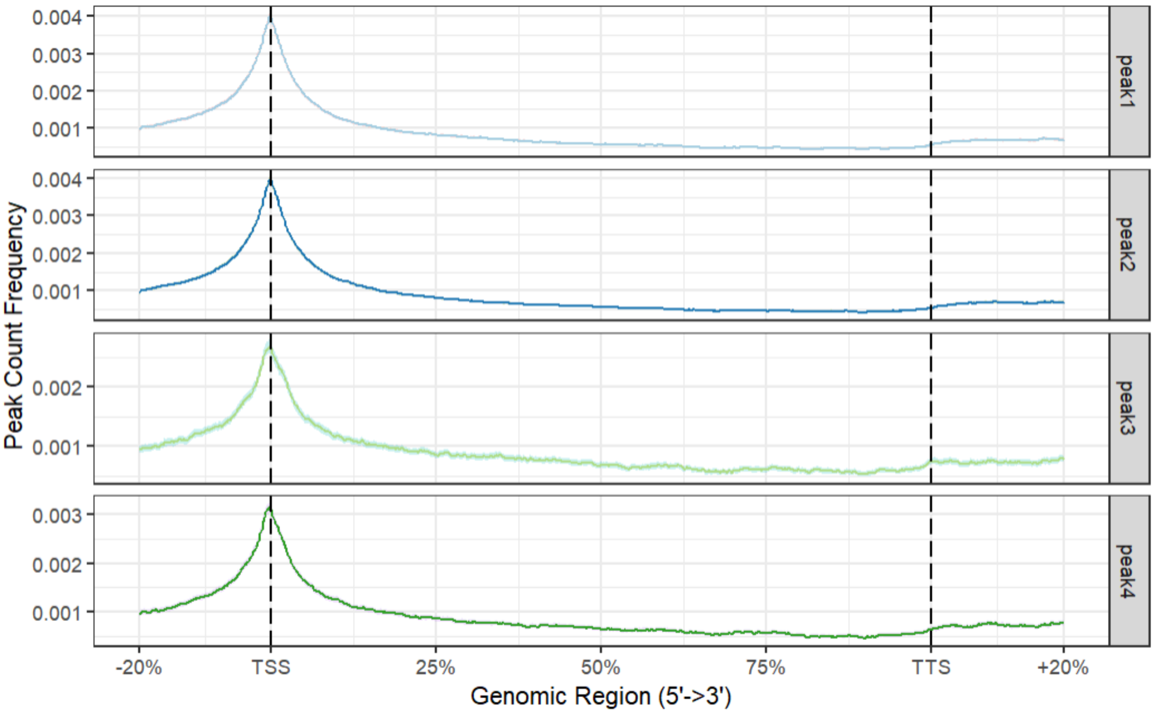
```
plotDistToTSS(peakAnno,title="Distribution of transcription factor-binding loci\nrelative to TSS")
```



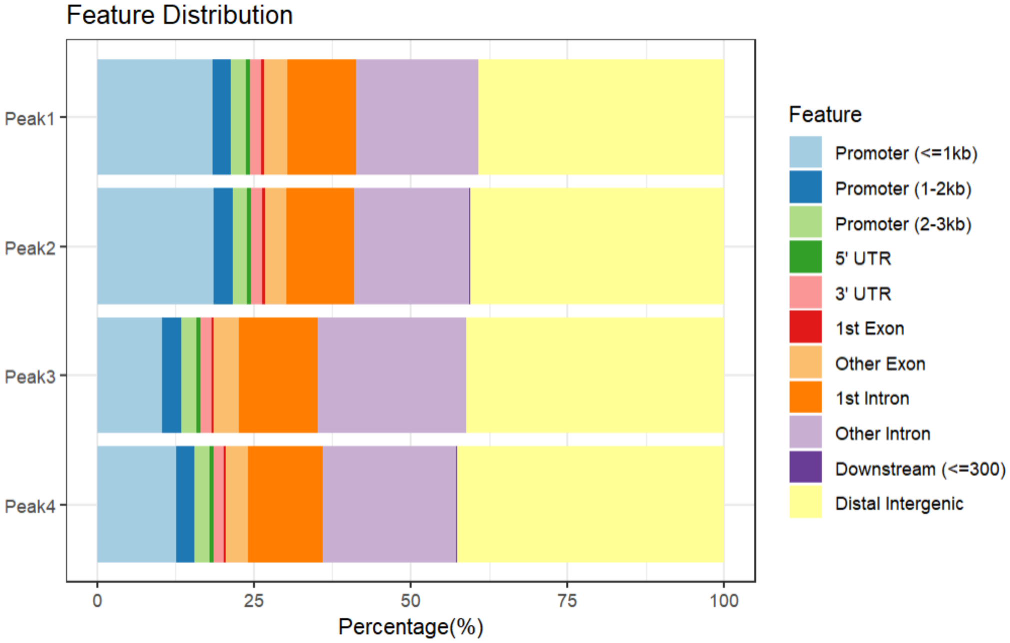
```
names(files) <- c("crELF3","wtELF3","crFOXA1","wtFOXA1")
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
txdb <- TxDb.Hsapiens.UCSC.hg19.knownGene
library(clusterProfiler)
promoter <- getPromoters(TxDb=txdb, upstream=3000, downstream=3000)
tagMatrixList <- lapply(Peaks,getTagMatrix, windows=promoter)
plotAvgProf(tagMatrixList, xlim=c(-3000, 3000),
           conf=0.95,resample=500, facet="row")
```



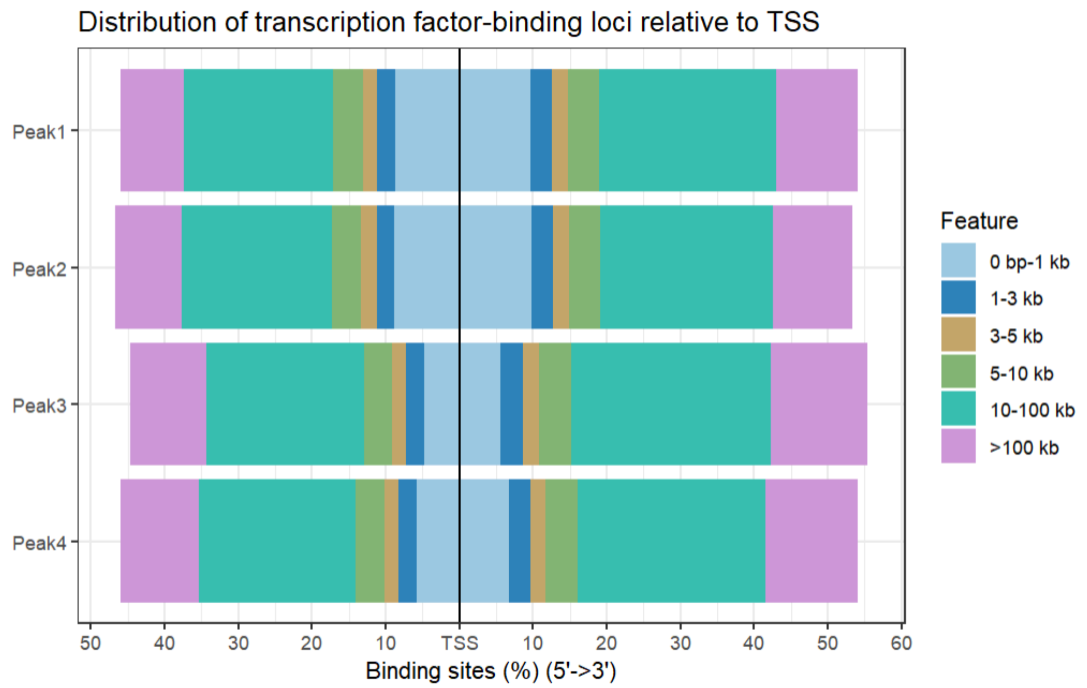
```
plotPeakProf2(Peaks, upstream = rel(0.2), downstream = rel(0.2),conf = 0.95, by = "gene", type =
"body",TxDb = txdb, facet = "row", nbin = 800)
```



```
peakAnnoList <- lapply(Peaks, annotatePeak, TxDb=txdb,
 tssRegion=c(-3000, 3000), verbose=FALSE)
plotAnnoBar(peakAnnoList)
```



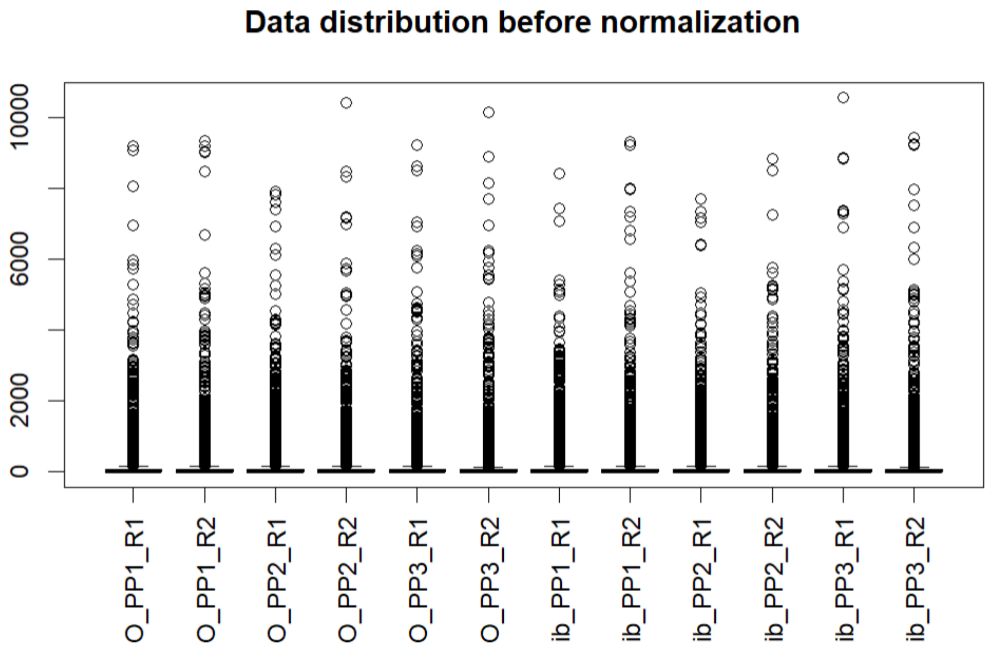
```
plotDistToTSS(peakAnnoList)
```



Distribution of transcription factor-binding loci relative to TSS

Proteomics

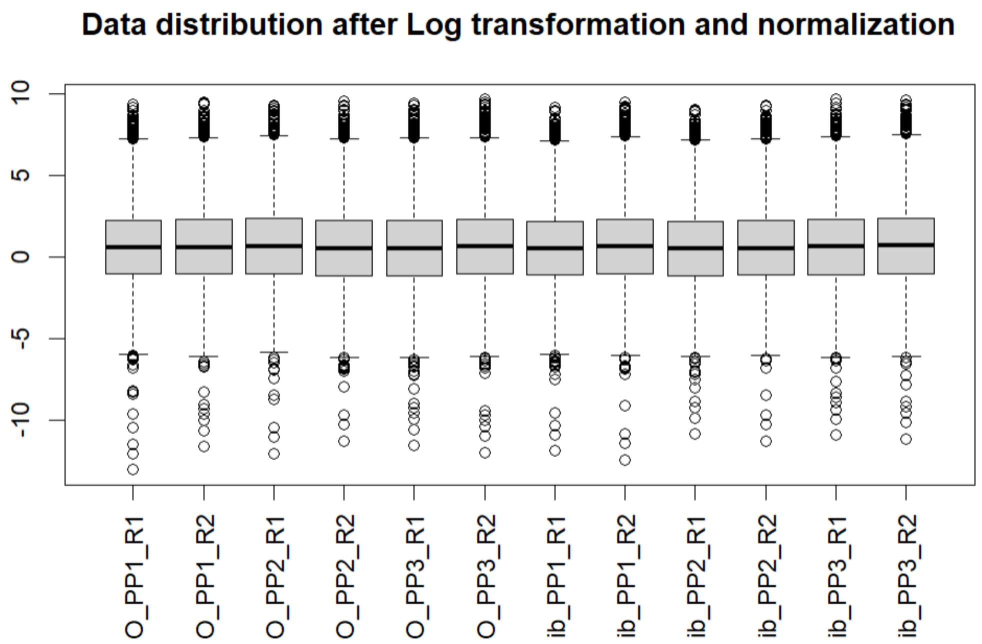
```
#LSD1i pretreated cells followwed by trametinib treatment
# Data distribution before normalization
boxplot(df.phosphoprot_assign[,2:ncol(df.phosphoprot_assign)],las=3, main="Data distribution before normalization")
```



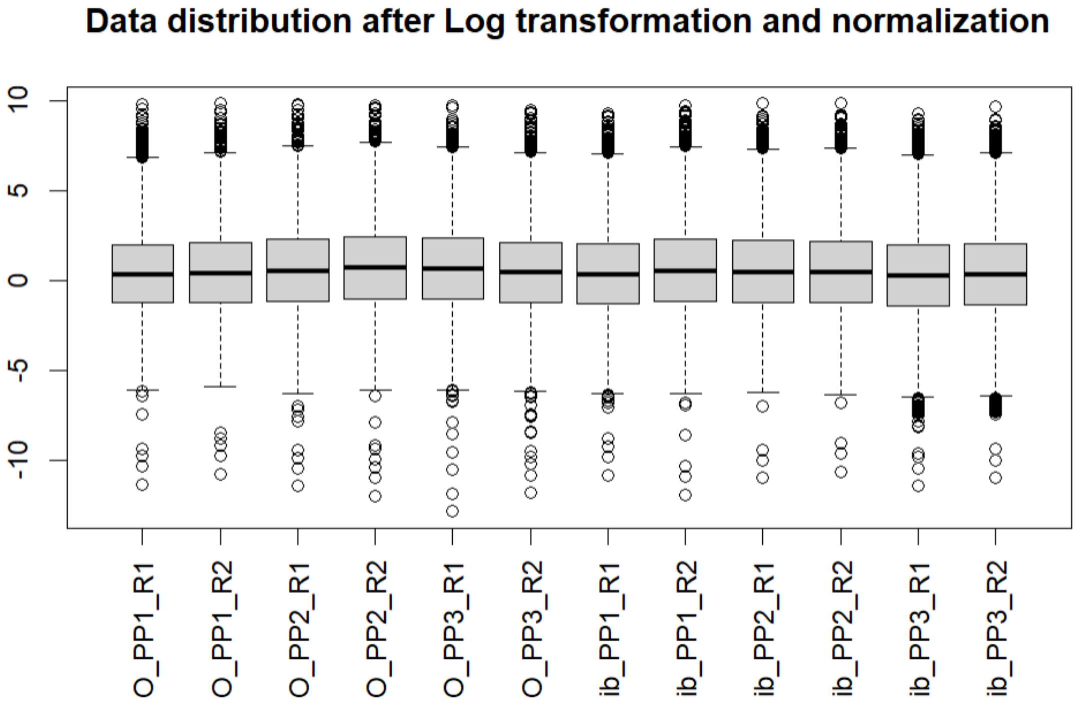
Data distribution before normalization

```
df.phosphoprot_assign <- data.frame(df.phosphoprot_assign)
# Log2 transform and normalize data by centering without scaling
df.norm_assign <- data.frame(site=df.phosphoprot_assign$site,
              scale(log2(df.phosphoprot_assign[,2:ncol(df.phosphoprot_assign)]), scale = F))

# Data distribution after normalization
```

```
boxplot(df.norm_assign[,2:ncol(df.norm_assign)],las=3, main="Data distribution after Log
transformation and normalization")
```



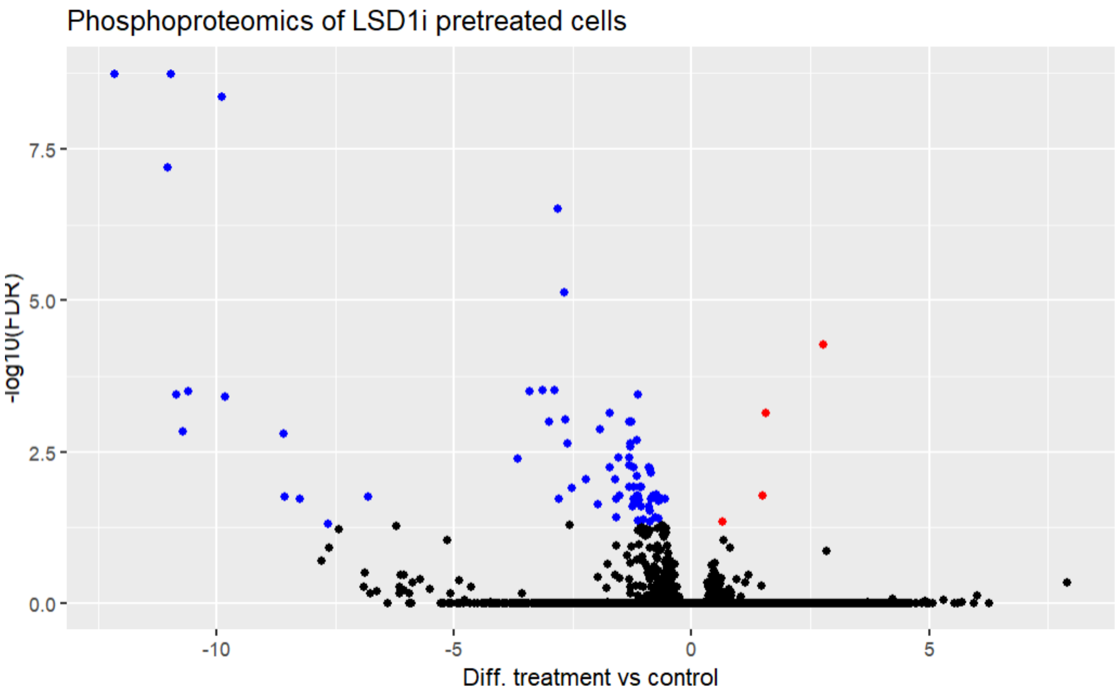
```
df.phosphoprot_2 <- data.frame(df.phosphoprot_2)
# Log2 transform and normalize data by centering without scaling
df.norm_2 <- data.frame(site=df.phosphoprot_2$site,
                scale(log2(df.phosphoprot_2[,2:ncol(df.phosphoprot_2)]), scale = F))
# Data distribution after normalization
boxplot(df.norm_2[,2:ncol(df.norm_2)],las=3, main="Data distribution after Log transformation and
normalization")
```



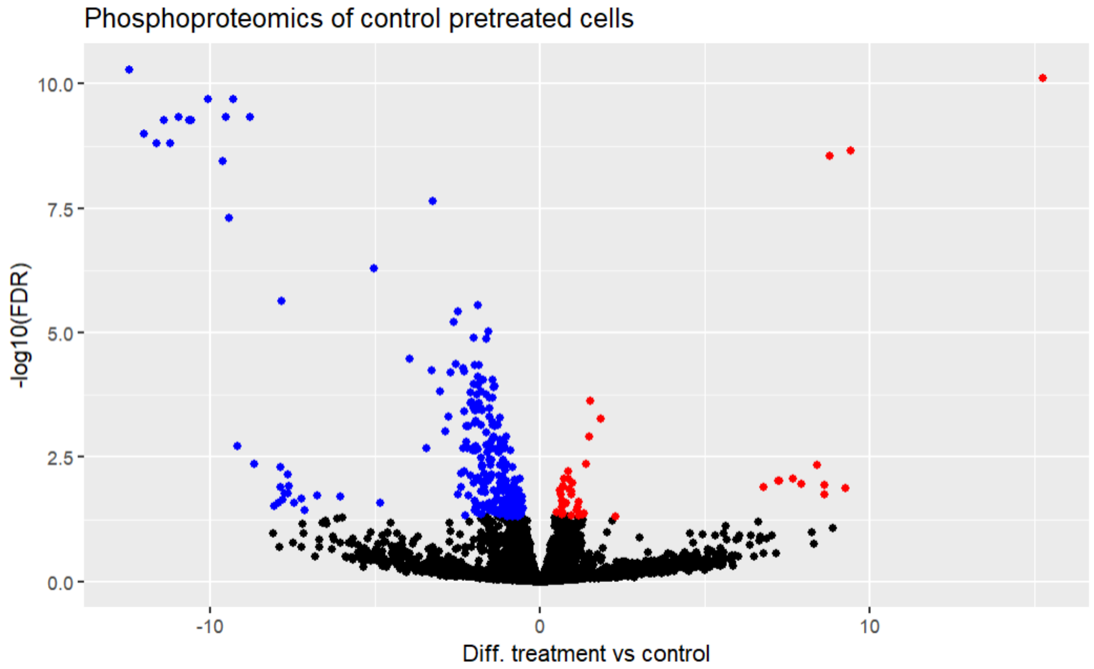
```
#LSD1i pretreated cells followwed by trametinib treatment
control_samples <- colnames(df.phosphoprot_assign[,2:7])
test_samples <- colnames(df.phosphoprot_assign[,8:13])

df.limma.results_assign <- protools2::compare.by.limma(df.to.compare = df.norm_assign,
                                        control.samples = control_samples,
                                        test.samples = test_samples)

write.csv(df.limma.results_assign, "Results of limma analysis of phopho.csv", row.names =
F)

head(df.limma.results_assign[order(df.limma.results_assign$difference.test.vs.control),])

df.significant_assign <- subset(df.limma.results_assign,df.limma.results_assign$FDR<0.05)
df.significant.decreased_assign <-
subset(df.significant_assign,df.significant_assign$difference.test.vs.control<0)
df.significant.increased_assign <-
subset(df.significant_assign,df.significant_assign$difference.test.vs.control>0)

nrow(df.significant.decreased_assign)
nrow(df.significant.increased_assign)

#control pretreated cells followwed by trametinib treatment
control_samples <- colnames(df.phosphoprot_2[,2:7])
test_samples <- colnames(df.phosphoprot_2[,8:13])

df.limma.results_2 <- protools2::compare.by.limma(df.to.compare = df.norm_2,
                                        control.samples = control_samples,
                                        test.samples = test_samples)
write.csv(df.limma.results_2, "Results of limma analysis of phopho.csv", row.names = F)

head(df.limma.results_2[order(df.limma.results_2$difference.test.vs.control),])

df.significant_2 <- subset(df.limma.results_2,df.limma.results_2$FDR<0.05)
df.significant.decreased_2 <-
subset(df.significant_2,df.significant_2$difference.test.vs.control<0)
df.significant.increased_2 <-
subset(df.significant_2,df.significant_2$difference.test.vs.control>0)

nrow(df.significant.decreased_2)
nrow(df.significant.increased_2)
```

Plot

```
library(ggplot2)
#LSD1i pretreated cells followwed by trametinib treatment
plot_assign <- ggplot(df.limma.results_assign,aes(x=difference.test.vs.control,y=-log10(FDR)))+
  geom_point()+
  geom_point(data=df.significant.decreased_assign,
             aes(x=difference.test.vs.control,y=-log10(FDR)),color="blue")+
  geom_point(data=df.significant.increased_assign,
             aes(x=difference.test.vs.control,y=-log10(FDR)),color="red")+
  labs(x="Diff. treatment vs control",
       title = "Phosphoproteomics of LSD1i pretreated cells")
plot_assign
```

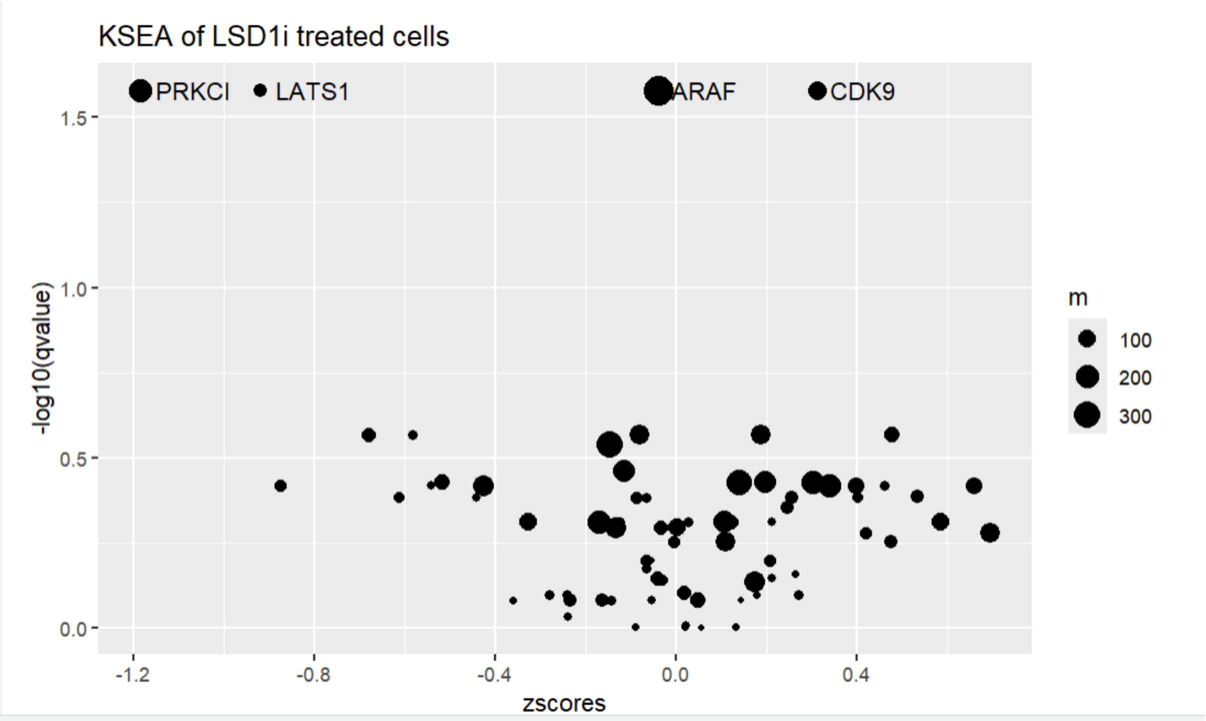


```
#control pretreated cells followwed by trametinib treatment
plot_2 <- ggplot(df.limma.results_2,aes(x=difference.test.vs.control,y=-log10(FDR)))+
  geom_point()+
  geom_point(data=df.significant.decreased_2,
             aes(x=difference.test.vs.control,y=-log10(FDR)),color="blue")+
  geom_point(data=df.significant.increased_2,
             aes(x=difference.test.vs.control,y=-log10(FDR)),color="red")+
  labs(x="Diff. treatment vs control",
       title = "Phosphoproteomics of control pretreated cells")
plot_2
```

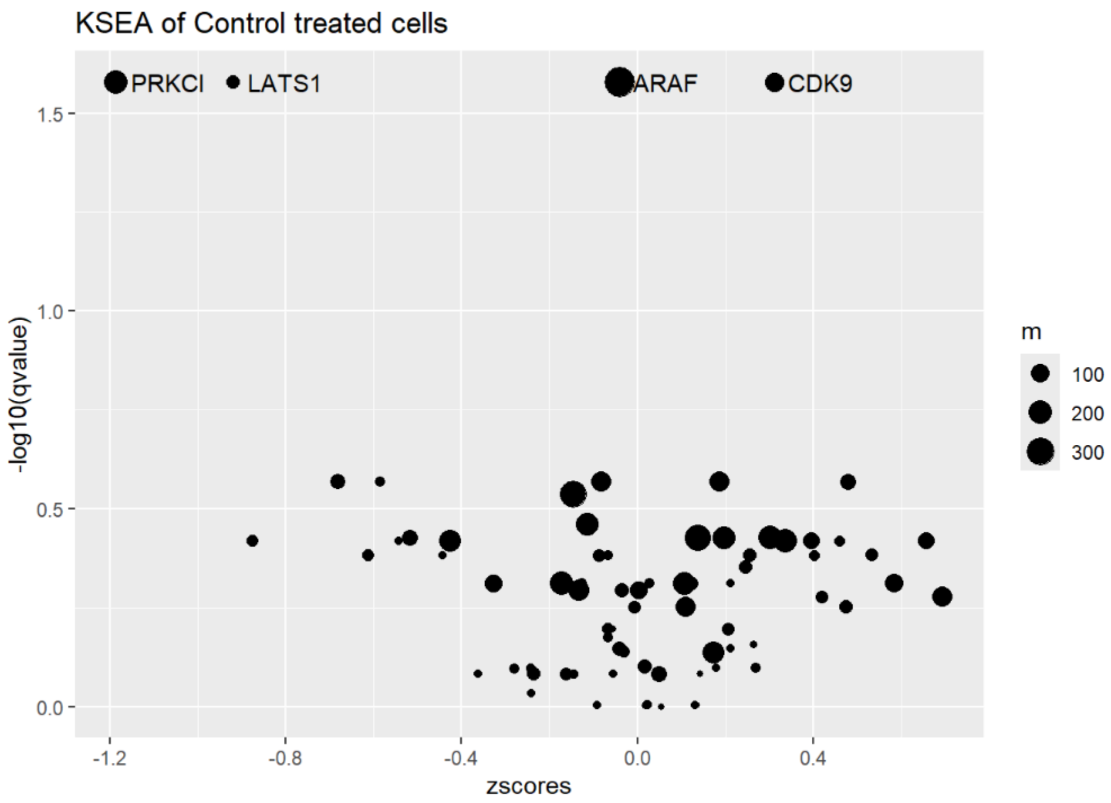


Volcano plot

```
#LSD1i pretreated cells followwed by trametinib treatment
assign_ksea <- read.csv("C:/Users/dell/Downloads/Data_2/Results of limma analysis of
phos.csv")
# Call the ksear.s function to carry out KSEA
#df.k <- ksear.s(data.frame(sites=df$site, df$differnce.test.vs.control),ks_db="pdts")
assign_ksea.k <- protools2::kinase.substrate.enrichment(dfx = assign_ksea,ks_db = "pdts")
# View the top kinases
assign_ksea.k[order(assign_ksea.k$pvalue),]

assign_ksea_1 <- subset(assign_ksea.k,assign_ksea.k$pvalues<.01)

plot.ksea.volcano <- ggplot(assign_ksea.k,aes(x=zscores,y=-log10(qvalue)))+
  geom_point(aes(size=m))+
  geom_text(data =
assign_ksea_1,aes(x=zscores,y=-log10(qvalue),label=kinases),hjust=-0.2)+
  labs(title = "KSEA of LSD1i treated cells")
plot.ksea.volcano
```



```
#Control pretreated cells followwed by trametinib treatment
assign_ksea_2 <-
read.csv("C:/Users/Downloads/OneDrive/Desktop/BCI/CAN7031_and_CAN7131_Omics_data_analytics_and_practical_training/Assignment/answers/limma_result_of_LSD1i_2.csv")
# Call the ksear.s function to carry out KSEA
#df.k <- ksear.s(data.frame(sites=df$site, df$differnce.test.vs.control),ks_db="pdts")
assign_ksea.k_2 <- protools2::kinase.substrate.enrichment(dfx = assign_ksea_2,ks_db =
"pdts")

assign_ksea.k_2[order(assign_ksea.k_2$pvalue),]

assign_ksea_2 <- subset(assign_ksea.k_2,assign_ksea.k_2$pvalues<.01)



plot.ksea.volcano_2 <- ggplot(assign_ksea.k_2,aes(x=zscores,y=-log10(qvalue)))+

  geom_point(aes(size=m))+

  geom_text(data =
assign_ksea_2,aes(x=zscores,y=-log10(qvalue),label=kinases),hjust=-0.2)+

  labs(title = "KSEA of Control treated cells")

plot.ksea.volcano_2
```



answer these questions:

1) Name the top 10 phosphorylation sites decreased and increased by the kinase inhibitor in each dataset

2) Name the top kinases increased by treatment in each dataset

3) Name the top kinases decreased by treatment in each dataset

4) What impact did the pre-treatment with the LSD1 inhibitor had in the effect of trametinib in decreasing kinase activities?

```
1) df.significant_assign$protein[1:10]
 [1] NCAPD2(S1330);NCAPD2(T1331)
 [2]NUP153(S522
 [3]ABL2(T938)
 [4]MYCBP2(T3470)
 [5]MAPK1(Y187)
 [6] MAPK1(T185)
 [7]NUP50(T219)
 [8] OPTN(S528)
 [9] LSP1(T184)
 [10]OSTF1(S213)
```

2)Top Kinases increased by treatment in each dataset:

#LSD1i pretreated cells followed by trametinib treatment

```
[1] IRAK1
[2] CAMKK2
[3] TAOK3
```

[4] CIT

#control pretreated cells followed by trametinib treatment
[1]CDK9
3) Top Kinases decreased by treatment in each dataset:
#LSD1i pretreated cells followed by trametinib treatment
[1] MAP2K1
[2] MAPK3
[3] TNK2
[4] MAPK1
#control pretreated cells followed by trametinib treatment
[1]PRKC1
[2] LATS1
[3] ARAF

4) In comparison between LSD1i pretreated cells to control pretreated cells, there were more kinases increased by treatment in LSD1i pretreated cells. In regards to kinases decreased by treatment, kinases in control pretreated cells were decreased more in comparison to the kinases in LSD1i pretreated cells.