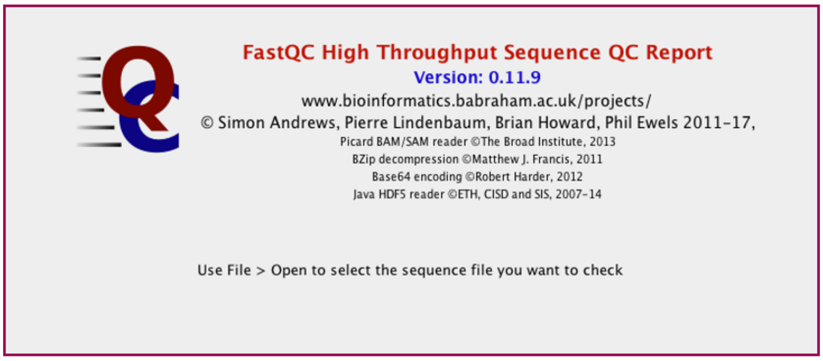
RNA Seq data analysis

For this analysis use the data sets (fast format) - Gr_11, Gr_12, Gr_13, Gr_31, Gr_32 and Gr_33 (Gr_11, Gr_12 and Gr_13 belong to the 1st group, whereas Gr_31, Gr_32 and Gr_33 belong to the other group)

1. Quality check of sequences

- This is done in FastQC
- **Command for running FastQC** - `fastqc *.fastq` **(in the terminal window)**



FastQC High Throughput Sequence QC Report

Version: 0.11.9

www.bioinformatics.babraham.ac.uk/projects/

© Simon Andrews, Pierre Lindenbaum, Brian Howard, Phil Ewels 2011-17,

Picard BAM/SAM reader ©The Broad Institute, 2013

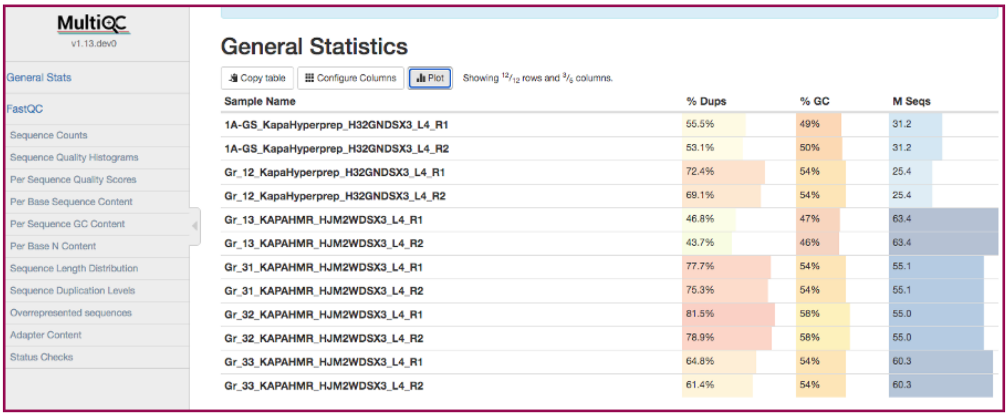
BZip decompression ©Matthew J. Francis, 2011

Base64 encoding ©Robert Harder, 2012

Java HDF5 reader ©ETH, CISD and SIS, 2007-14

Use File > Open to select the sequence file you want to check

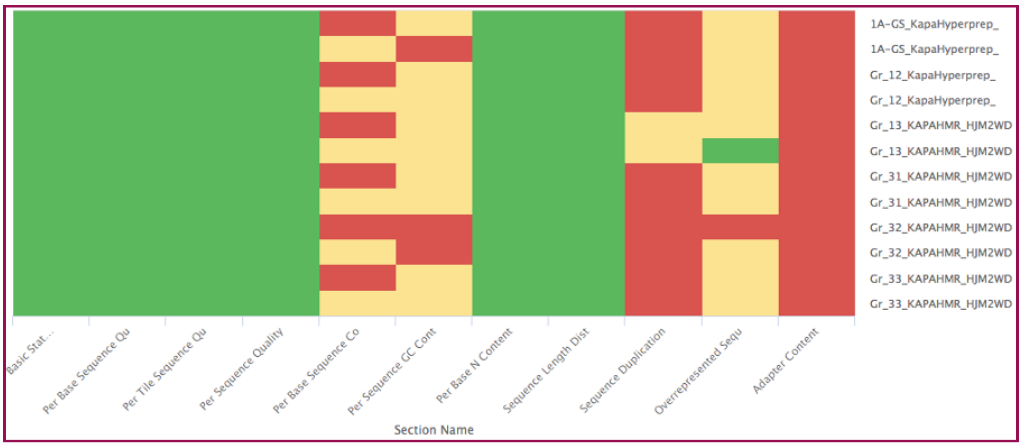
- FastQC will generate a html and a zip file. The html file of each of the fastq files gives a detailed report of each fastq file.
- Running MultiQC a single summary report to visualise the combined results across all samples can be created
  - **Command for running MultiQC** - `multiqc .` **(multiqc space and dot) - It** creates a multiqc_report.html file

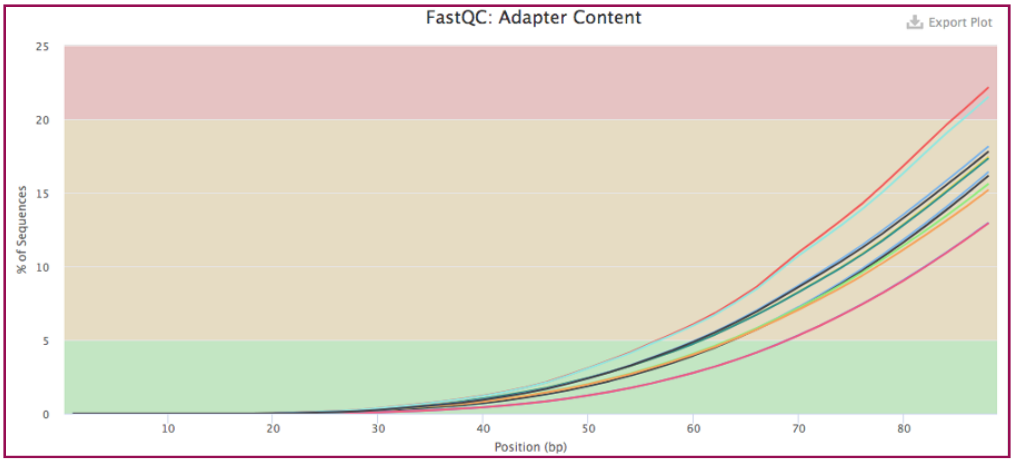

MultiQC
v1.13.dev0
General Stats
FastQC
Sequence Counts
Sequence Quality Histograms
Per Sequence Quality Scores
Per Base Sequence Content
Per Sequence GC Content
Per Base N Content
Sequence Length Distribution
Sequence Duplication Levels
Overrepresented sequences
Adapter Content
Status Checks
Copy table
Configure Columns
Plot

General Statistics

Showing $^{12}/_{12}$ rows and $^{3}/_{5}$ columns.

Sample Name	% Dups	% GC	M Seqs
1A-GS_KapaHyperprep_H32GNDSX3_L4_R1	55.5%	49%	31.2
1A-GS_KapaHyperprep_H32GNDSX3_L4_R2	53.1%	50%	31.2
Gr_12_KapaHyperprep_H32GNDSX3_L4_R1	72.4%	54%	25.4
Gr_12_KapaHyperprep_H32GNDSX3_L4_R2	69.1%	54%	25.4
Gr_13_KAPAHMR_HJM2WDSX3_L4_R1	46.8%	47%	63.4
Gr_13_KAPAHMR_HJM2WDSX3_L4_R2	43.7%	46%	63.4
Gr_31_KAPAHMR_HJM2WDSX3_L4_R1	77.7%	54%	55.1
Gr_31_KAPAHMR_HJM2WDSX3_L4_R2	75.3%	54%	55.1
Gr_32_KAPAHMR_HJM2WDSX3_L4_R1	81.5%	58%	55.0
Gr_32_KAPAHMR_HJM2WDSX3_L4_R2	78.9%	58%	55.0
Gr_33_KAPAHMR_HJM2WDSX3_L4_R1	64.8%	54%	60.3
Gr_33_KAPAHMR_HJM2WDSX3_L4_R2	61.4%	54%	60.3





According to the overall statistics, per base quality is good in all the samples, so there is no requirement of filtering the low quality reads. Whereas, the adapter content in all the samples is bad.

2. Removal of adapters - use cutadapt

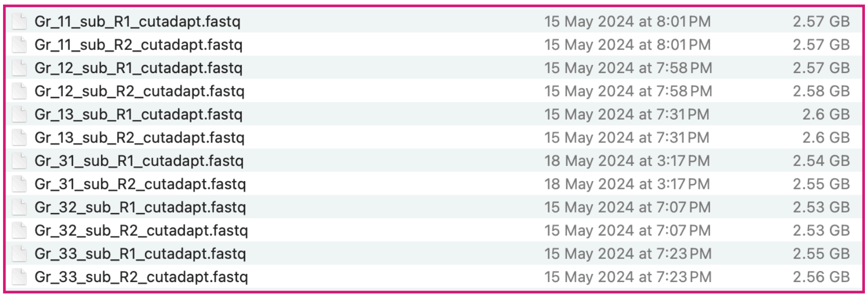
The list of adapters can be obtained along with the sequencing data.

Here, the adapters used are - `AGATCGGAAGAGCACACGTCTGAACTCCAGTCA` and `AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT`

Command for running Cutadapt

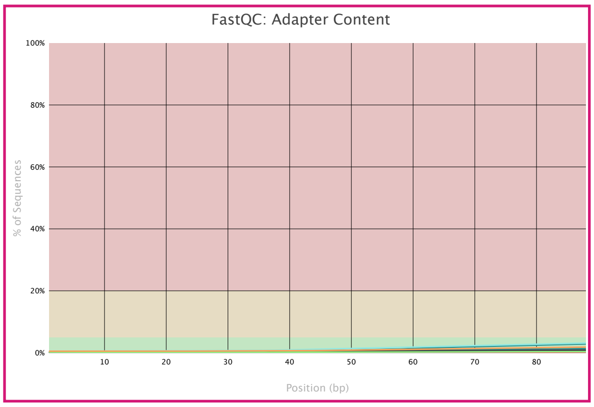
```
cutadapt -b AGATCGGAAGAGCACACGTCTGAACTCCAGTCA -B
AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT -o Gr_32_sub_R1_cutadapt.fastq
-p Gr_32_sub_R2_cutadapt.fastq Gr_32_sub_R1.fastq
Gr_32_sub_R2.fastq
```

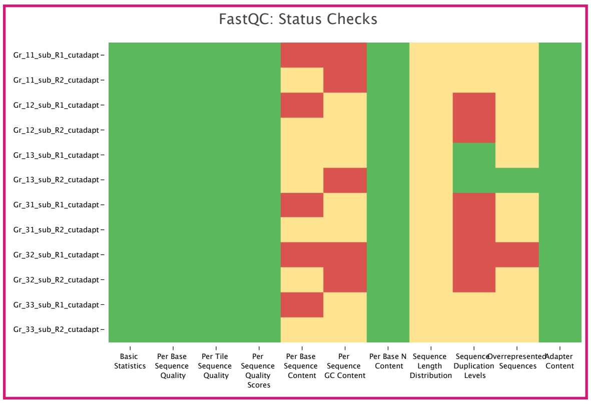
Use the above command for all the paired end fastq files to generate files as below



Gr_11_sub_R1_cutadapt.fastq	15 May 2024 at 8:01 PM	2.57 GB
Gr_11_sub_R2_cutadapt.fastq	15 May 2024 at 8:01 PM	2.57 GB
Gr_12_sub_R1_cutadapt.fastq	15 May 2024 at 7:58 PM	2.57 GB
Gr_12_sub_R2_cutadapt.fastq	15 May 2024 at 7:58 PM	2.58 GB
Gr_13_sub_R1_cutadapt.fastq	15 May 2024 at 7:31 PM	2.6 GB
Gr_13_sub_R2_cutadapt.fastq	15 May 2024 at 7:31 PM	2.6 GB
Gr_31_sub_R1_cutadapt.fastq	18 May 2024 at 3:17 PM	2.54 GB
Gr_31_sub_R2_cutadapt.fastq	18 May 2024 at 3:17 PM	2.55 GB
Gr_32_sub_R1_cutadapt.fastq	15 May 2024 at 7:07 PM	2.53 GB
Gr_32_sub_R2_cutadapt.fastq	15 May 2024 at 7:07 PM	2.53 GB
Gr_33_sub_R1_cutadapt.fastq	15 May 2024 at 7:23 PM	2.55 GB
Gr_33_sub_R2_cutadapt.fastq	15 May 2024 at 7:23 PM	2.56 GB

Note :- Go for FastQC with the new files and check for quality again - proceed only after the quality is good



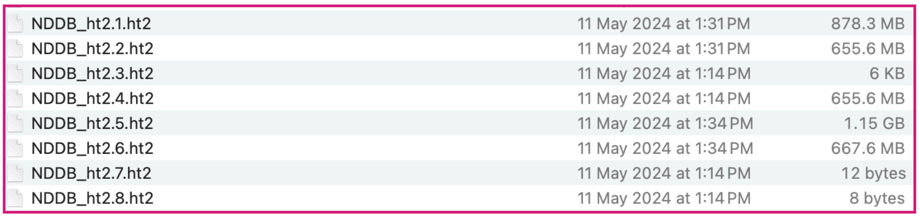


3. **Mapping to the reference using an aligner -** Here we use hisat2

   1. **Build the index of the NDDB_SH_1_genome.fna genome :-**

      - **Command -** `hisat2-build /Users/rk_gandham_shree/Downloads/rsem/10Million/NDDB_SH_1_genome.fna NDDB_ht2`

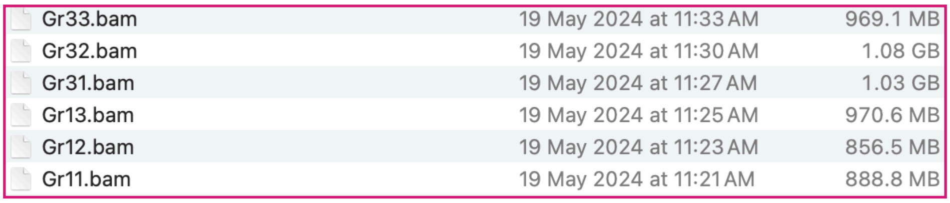
        - **It builds an index with the prefix** `NDDB_ht2`



NDDB_ht2.1.ht2	11 May 2024 at 1:31 PM	878.3 MB
NDDB_ht2.2.ht2	11 May 2024 at 1:31 PM	655.6 MB
NDDB_ht2.3.ht2	11 May 2024 at 1:14 PM	6 KB
NDDB_ht2.4.ht2	11 May 2024 at 1:14 PM	655.6 MB
NDDB_ht2.5.ht2	11 May 2024 at 1:34 PM	1.15 GB
NDDB_ht2.6.ht2	11 May 2024 at 1:34 PM	667.6 MB
NDDB_ht2.7.ht2	11 May 2024 at 1:14 PM	12 bytes
NDDB_ht2.8.ht2	11 May 2024 at 1:14 PM	8 bytes

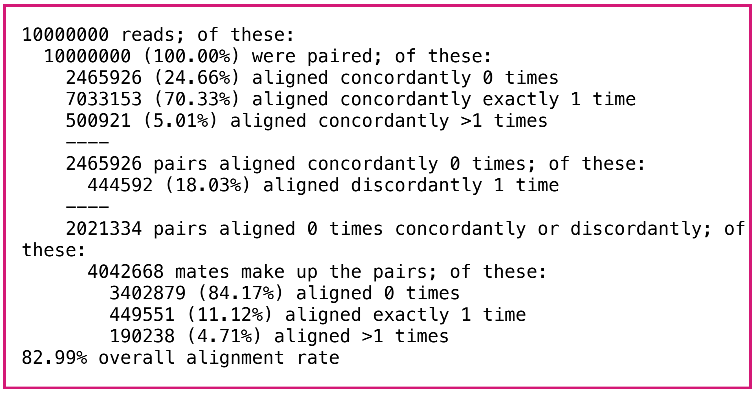
   2. **Map the reads to the reference :-**

      - **Command -** `hisat2 -p 6 -x NDDB_ht2 -1 Gr_11_sub_R1_cutadapt.fastq -2 Gr_11_sub_R2_cutadapt.fastq | samtools sort -@ 6 -o Gr11.bam_new ; hisat2 -p 6 -x NDDB_ht2 -1 Gr_12_sub_R1_cutadapt.fastq -2 Gr_12_sub_R2_cutadapt.fastq | samtools sort -@ 6 -o Gr12.bam_new ; hisat2 -p 6 -x NDDB_ht2 -1 Gr_13_sub_R1_cutadapt.fastq -2 Gr_13_sub_R2_cutadapt.fastq | samtools sort -@ 6 -o Gr13.bam_new ; hisat2 -p 6 -x NDDB_ht2 -1 Gr_31_sub_R1_cutadapt.fastq -2 Gr_31_sub_R2_cutadapt.fastq | samtools sort -@ 6 -o Gr31.bam_new ; hisat2 -p 6 -x NDDB_ht2 -1 Gr_32_sub_R1_cutadapt.fastq -2 Gr_32_sub_R2_cutadapt.fastq | samtools sort -@ 6 -o Gr32.bam_new ; hisat2 -p 6 -x NDDB_ht2 -1 Gr_33_sub_R1_cutadapt.fastq -2 Gr_33_sub_R2_cutadapt.fastq | samtools sort -@ 6 -o Gr33.bam_new`



Gr33.bam	19 May 2024 at 11:33 AM	969.1 MB
Gr32.bam	19 May 2024 at 11:30 AM	1.08 GB
Gr31.bam	19 May 2024 at 11:27 AM	1.03 GB
Gr13.bam	19 May 2024 at 11:25 AM	970.6 MB
Gr12.bam	19 May 2024 at 11:23 AM	856.5 MB
Gr11.bam	19 May 2024 at 11:21 AM	888.8 MB

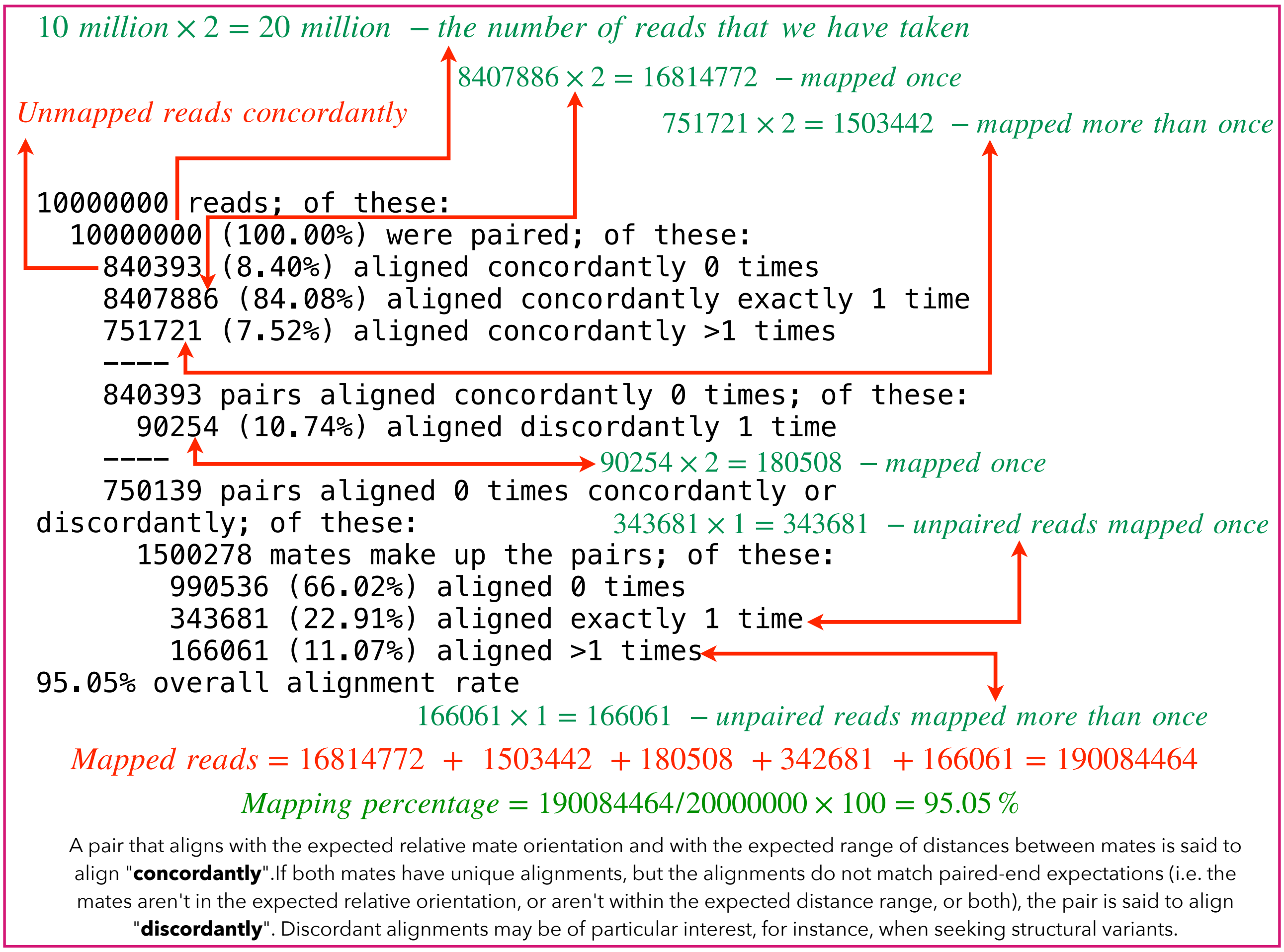
Note : The mapping quality improved after removing the adapters - Below is before removing



```
10000000 reads; of these:
  10000000 (100.00%) were paired; of these:
    2465926 (24.66%) aligned concordantly 0 times
    7033153 (70.33%) aligned concordantly exactly 1 time
    500921 (5.01%) aligned concordantly >1 times
    ----
    2465926 pairs aligned concordantly 0 times; of these:
      444592 (18.03%) aligned discordantly 1 time
    ----
    2021334 pairs aligned 0 times concordantly or discordantly; of
these:
      4042668 mates make up the pairs; of these:
        3402879 (84.17%) aligned 0 times
        449551 (11.12%) aligned exactly 1 time
        190238 (4.71%) aligned >1 times
82.99% overall alignment rate
```

Before removing the adapters the mapping percentage was 82.99% and after removing the adapters the mapping percentage increased to 95.05%

Mapping quality after removal of adapters is given below:



$10\ million \times 2 = 20\ million$ *– the number of reads that we have taken*

$8407886 \times 2 = 16814772$ *– mapped once*

Unmapped reads concordantly

$751721 \times 2 = 1503442$ *– mapped more than once*

```
10000000 reads; of these:
  10000000 (100.00%) were paired; of these:
    840393 (8.40%) aligned concordantly 0 times
    8407886 (84.08%) aligned concordantly exactly 1 time
    751721 (7.52%) aligned concordantly >1 times
    ----
    840393 pairs aligned concordantly 0 times; of these:
      90254 (10.74%) aligned discordantly 1 time
    ----
    750139 pairs aligned 0 times concordantly or
discordantly; of these:
      1500278 mates make up the pairs; of these:
        990536 (66.02%) aligned 0 times
        343681 (22.91%) aligned exactly 1 time
        166061 (11.07%) aligned >1 times
95.05% overall alignment rate
```

$90254 \times 2 = 180508$ *– mapped once*

$343681 \times 1 = 343681$ *– unpaired reads mapped once*

$166061 \times 1 = 166061$ *– unpaired reads mapped more than once*

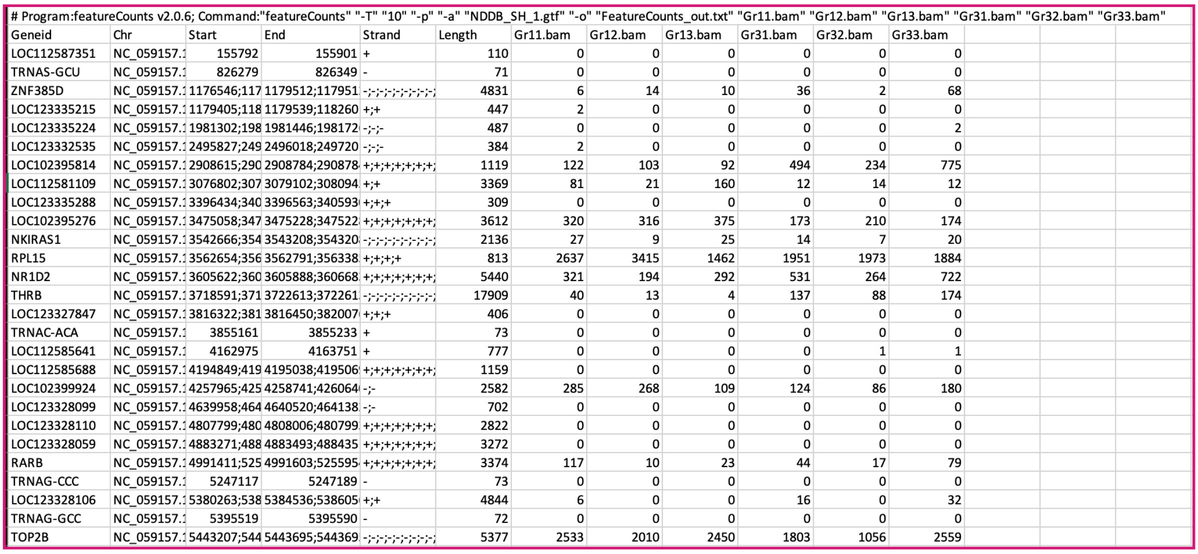
$$Mapped\ reads = 16814772\ +\ 1503442\ + 180508\ + 342681\ + 166061 = 190084464$$

$$Mapping\ percentage = 190084464/20000000 \times 100 = 95.05\ \%$$

A pair that aligns with the expected relative mate orientation and with the expected range of distances between mates is said to align "**concordantly**".If both mates have unique alignments, but the alignments do not match paired-end expectations (i.e. the mates aren't in the expected relative orientation, or aren't within the expected distance range, or both), the pair is said to align "**discordantly**". Discordant alignments may be of particular interest, for instance, when seeking structural variants.

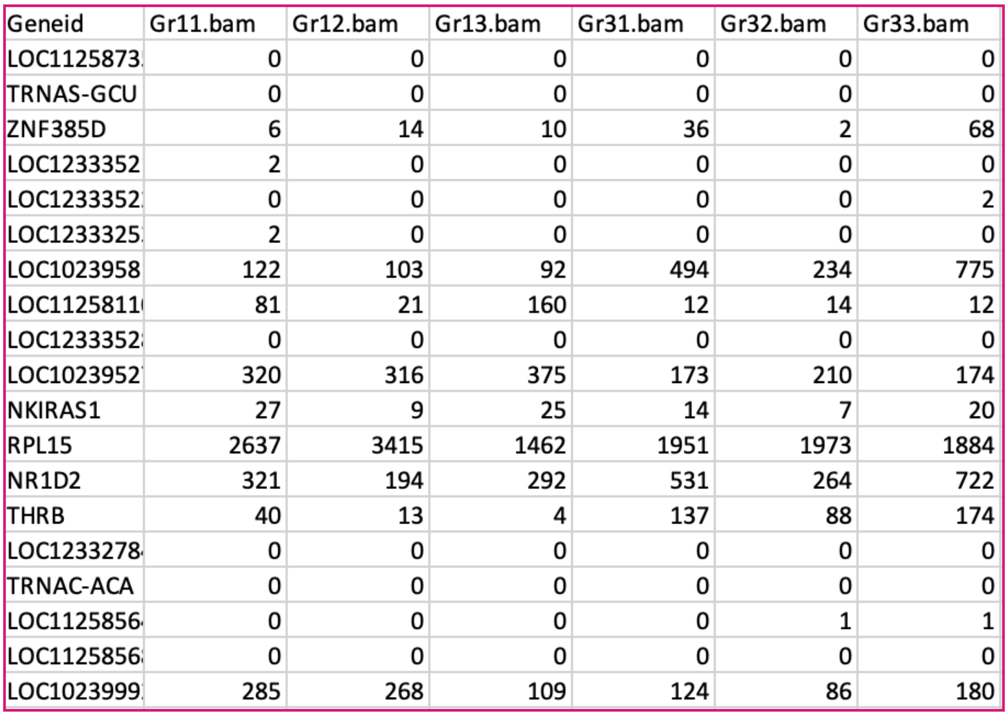
4. Getting the counts from the bam file - Here we use featureCounts

- **Command -** `./featureCounts -T 10 -p -a NDDB_SH_1.gtf -o FeatureCounts_out.txt Gr11.bam Gr12.bam Gr13.bam Gr31.bam Gr32.bam Gr33.bam`
- `FeatureCounts_out.txt opened in excel is given below`



# Program:featureCounts v2.0.6; Command:"featureCounts" "-T" "10" "-p" "-a" "NDDB_SH_1.gtf" "-o" "FeatureCounts_out.txt" "Gr11.bam" "Gr12.bam" "Gr13.bam" "Gr31.bam" "Gr32.bam" "Gr33.bam"											
Geneid	Chr	Start	End	Strand	Length	Gr11.bam	Gr12.bam	Gr13.bam	Gr31.bam	Gr32.bam	Gr33.bam
LOC112587351	NC_059157.1	155792	155901	+	110	0	0	0	0	0	0
TRNAS-GCU	NC_059157.1	826279	826349	-	71	0	0	0	0	0	0
ZNF385D	NC_059157.1	1176546;117	1179512;117951	-;-;-;-;-;-;-;-;-;	4831	6	14	10	36	2	68
LOC123335215	NC_059157.1	1179405;118	1179539;118260	+;+	447	2	0	0	0	0	0
LOC123335224	NC_059157.1	1981302;198	1981446;198172	-;-;-	487	0	0	0	0	0	2
LOC123332535	NC_059157.1	2495827;249	2496018;249720	-;-;-	384	2	0	0	0	0	0
LOC102395814	NC_059157.1	2908615;290	2908784;290878	+;+;+;+;+;+;+;	1119	122	103	92	494	234	775
LOC112581109	NC_059157.1	3076802;307	3079102;308094	+;+	3369	81	21	160	12	14	12
LOC123335288	NC_059157.1	3396434;340	3396563;340593	+;+;+	309	0	0	0	0	0	0
LOC102395276	NC_059157.1	3475058;347	3475228;347522	+;+;+;+;+;+;+;	3612	320	316	375	173	210	174
NKIRAS1	NC_059157.1	3542666;354	3543208;354320	-;-;-;-;-;-;-;-;-;	2136	27	9	25	14	7	20
RPL15	NC_059157.1	3562654;356	3562791;356338	+;+;+;+	813	2637	3415	1462	1951	1973	1884
NR1D2	NC_059157.1	3605622;360	3605888;360668	+;+;+;+;+;+;+;	5440	321	194	292	531	264	722
THRB	NC_059157.1	3718591;371	3722613;372261	-;-;-;-;-;-;-;-;-;	17909	40	13	4	137	88	174
LOC123327847	NC_059157.1	3816322;381	3816450;382007	+;+;+	406	0	0	0	0	0	0
TRNAC-ACA	NC_059157.1	3855161	3855233	+	73	0	0	0	0	0	0
LOC112585641	NC_059157.1	4162975	4163751	+	777	0	0	0	0	1	1
LOC112585688	NC_059157.1	4194849;419	4195038;419506	+;+;+;+;+;+;+;	1159	0	0	0	0	0	0
LOC102399924	NC_059157.1	4257965;425	4258741;426064	-;-	2582	285	268	109	124	86	180
LOC123328099	NC_059157.1	4639958;464	4640520;464138	-;-	702	0	0	0	0	0	0
LOC123328110	NC_059157.1	4807799;480	4808006;480799	+;+;+;+;+;+;+;	2822	0	0	0	0	0	0
LOC123328059	NC_059157.1	4883271;488	4883493;488435	+;+;+;+;+;+;+;	3272	0	0	0	0	0	0
RARB	NC_059157.1	4991411;525	4991603;525595	+;+;+;+;+;+;+;	3374	117	10	23	44	17	79
TRNAG-CCC	NC_059157.1	5247117	5247189	-	73	0	0	0	0	0	0
LOC123328106	NC_059157.1	5380263;538	5384536;538605	+;+	4844	6	0	0	16	0	32
TRNAG-GCC	NC_059157.1	5395519	5395590	-	72	0	0	0	0	0	0
TOP2B	NC_059157.1	5443207;544	5443695;544369	-;-;-;-;-;-;-;-;-;	5377	2533	2010	2450	1803	1056	2559

- For differential we need the count in each sample for all the genes
- **Command -** `awk '{ print $1,  $7, $8, $9, $10, $11, $12 }' FeatureCounts_out.txt > FeatureCounts_final.txt`



Geneid	Gr11.bam	Gr12.bam	Gr13.bam	Gr31.bam	Gr32.bam	Gr33.bam
LOC1125873	0	0	0	0	0	0
TRNAS-GCU	0	0	0	0	0	0
ZNF385D	6	14	10	36	2	68
LOC1233352	2	0	0	0	0	0
LOC1233352	0	0	0	0	0	2
LOC1233325	2	0	0	0	0	0
LOC1023958	122	103	92	494	234	775
LOC1125811	81	21	160	12	14	12
LOC1233352	0	0	0	0	0	0
LOC1023952	320	316	375	173	210	174
NKIRAS1	27	9	25	14	7	20
RPL15	2637	3415	1462	1951	1973	1884
NR1D2	321	194	292	531	264	722
THRB	40	13	4	137	88	174
LOC1233278	0	0	0	0	0	0
TRNAC-ACA	0	0	0	0	0	0
LOC1125856	0	0	0	0	1	1
LOC1125856	0	0	0	0	0	0
LOC1023999	285	268	109	124	86	180

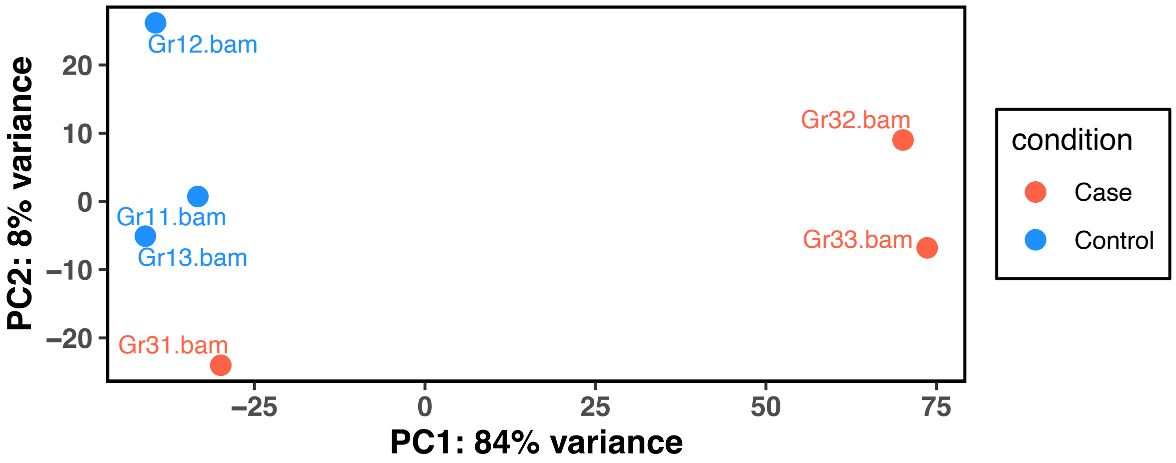
5. **After getting the counts -** do a PCA to see whether the samples cluster as per the requirement - the treatment samples are expected to cluster together and the control samples together

   R-Script for PCA

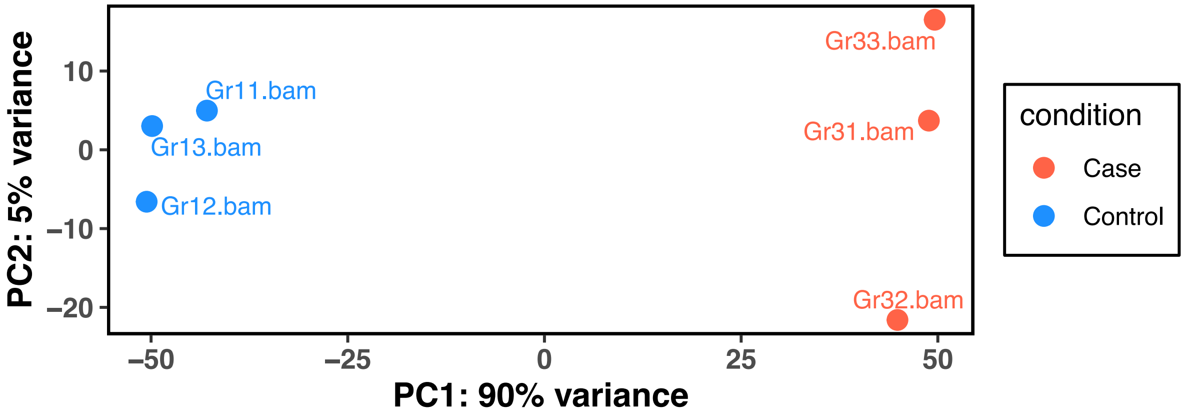
```
library(DESeq2)
library(ggplot2)
library(ggrepel)
pdf("dfn.pdf",width=6,height = 4)
counts <- read.table('FeatureCounts_final.txt', header = TRUE,
row.names = 1)
class(counts)
head(counts)
countdata <- data.matrix(counts)
class(countdata)
countdata <- round(countdata)
Design <- data.frame(
  row.names = colnames(countdata),
  condition = c("Control", "Control", "Control", "Case", "Case",
"Case"),
  libType = c("Single-end", "Single-end", "Single-end", "Single-
end", "Single-end", "Single-end"))
Design
dds <- DESeqDataSetFromMatrix(countData = countdata, colData =
Design, design = ~ condition)
dds
vsd <- vst(dds, blind=FALSE)
```

```
pcaData <- plotPCA(vsd, intgroup = c("condition"), returnData =
TRUE)
percentVar <- round(100 * attr(pcaData, "percentVar"))
custom_colors <- c("Control" = "dodgerblue", "Case" = "tomato")
pcaPlot <- ggplot(pcaData, aes(PC1, PC2, color = condition,
label = name)) +
  geom_point(size = 3) +
  geom_text_repel(size = 3) +
  scale_color_manual(values = custom_colors) +
  theme_classic() +
  xlab(paste0("PC1: ", percentVar[1], "% variance")) +
  ylab(paste0("PC2: ", percentVar[2], "% variance")) +
  coord_fixed() +theme(legend.position = "right",
      legend.background = element_rect(color = "black",
                                       fill = "white",
                                       size = 0.5,
                                       linetype =
"solid"),panel.border = element_rect(color = "black", fill = NA,
size = 1))
p<-pcaPlot+
theme(axis.title=element_text(size=12,face="bold"),axis.text.x =
element_text(size = 10,face="bold"),axis.text.y =
element_text(size = 10,face="bold"),axis.line = element_blank())
print(p)
dev.off()
```

The PCA for counts using the above script showed that G31 is clustering with Control (figure below). Therefore, this sample cannot be considered for further analysis.

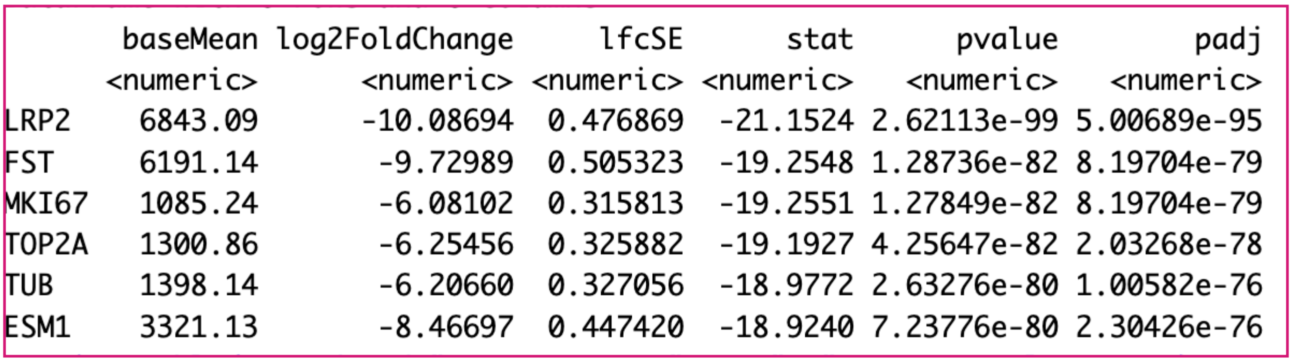


A different biological replicate was sequenced and counts for the same were obtained as above and the PCA was redone (Figure below). Now, the samples of case and control clearly clustered separately. The samples can be considered for further analysis



6. Differential expression using DESeq2 and EdgeR

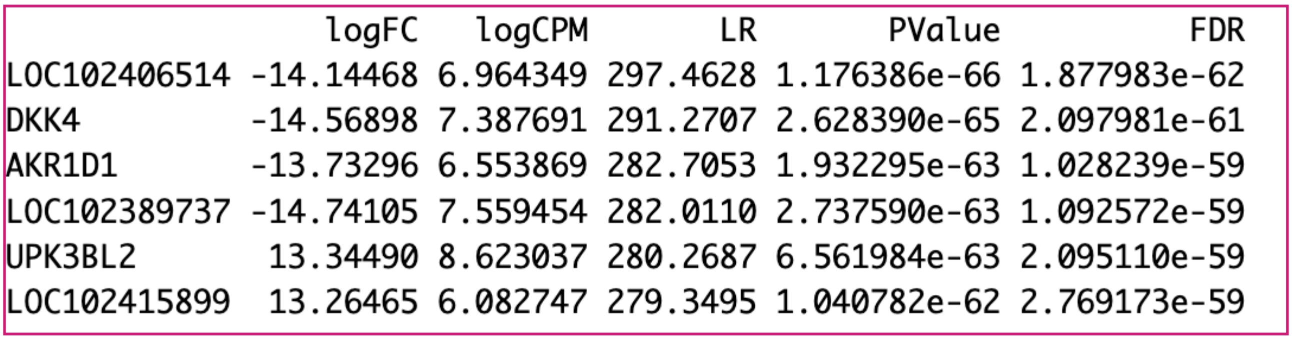
```
library(DESeq2)
counts <- read.table('FeatureCounts_final.txt', header=TRUE,
row.names=1)
class(counts)
head(counts)
countdata=data.matrix(counts)
class(countdata)
Design <- data.frame(row.names =colnames(counts), condition =
c("Control", "Control","Control", "Treated", "Treated","Treated"),
libType = c("paired-end", "paired-end","paired-end","paired-
end","paired-end", "paired-end"))
Design
dds <- DESeqDataSetFromMatrix(countData = round(countdata),colData
= Design,design = ~condition)
dds
dds <- DESeq(dds)
res <- results(dds)
resOrdered <- res[order(res$padj),]
head(resOrdered)
write.table(resOrdered,”
DESeq2_FC.txt",sep="\t",quote=F,col.names=T)
```



```
         baseMean log2FoldChange     lfcSE      stat      pvalue        padj
        <numeric>      <numeric> <numeric> <numeric>   <numeric>   <numeric>
LRP2      6843.09      -10.08694  0.476869  -21.1524 2.62113e-99 5.00689e-95
FST       6191.14       -9.72989  0.505323  -19.2548 1.28736e-82 8.19704e-79
MKI67     1085.24       -6.08102  0.315813  -19.2551 1.27849e-82 8.19704e-79
TOP2A     1300.86       -6.25456  0.325882  -19.1927 4.25647e-82 2.03268e-78
TUB       1398.14       -6.20660  0.327056  -18.9772 2.63276e-80 1.00582e-76
ESM1      3321.13       -8.46697  0.447420  -18.9240 7.23776e-80 2.30426e-76
```

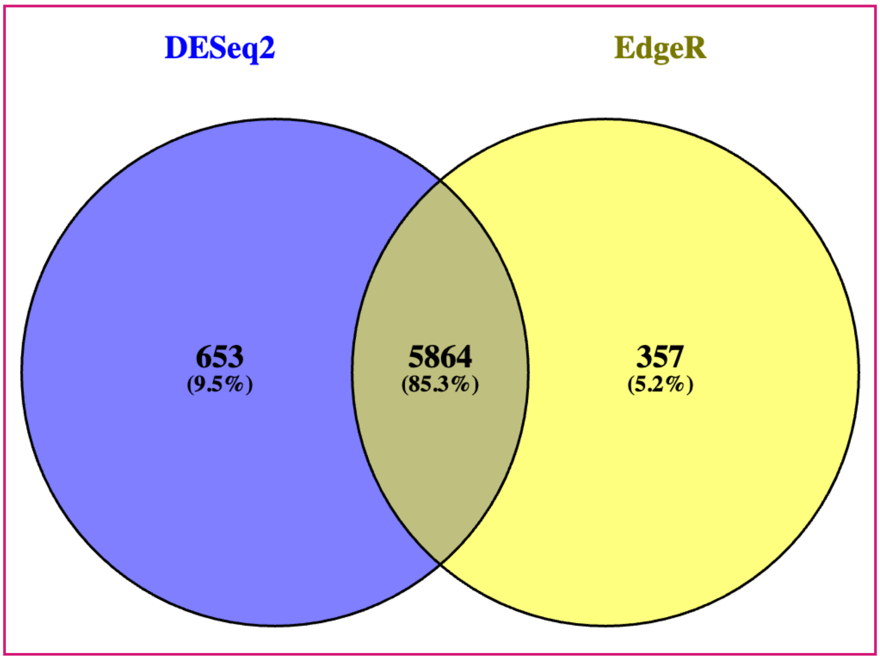
```
library(edgeR)

counts <- read.table('FeatureCounts_final.txt', header = TRUE,
row.names = 1)
countdata <- data.matrix(counts)
head(counts)
```

```
Design <- data.frame(row.names = colnames(counts), condition =
c("Control", "Control","Control", "Treated", "Treated","Treated"),
libType = c("paired-end", "paired-end","paired-end","paired-
end","paired-end", "paired-end"))
dge <- DGEList(counts = countdata, group = Design$condition)
keep <- rowSums(cpm(dge) > 1) >= 2
dge <- dge[keep,]
dge <- calcNormFactors(dge)
head(dge)
design <- model.matrix(~condition, data = Design)
dge <- estimateGLMCommonDisp(dge, design)
dge <- estimateGLMTrendedDisp(dge, design)
dge <- estimateGLMTagwiseDisp(dge, design)
fit <- glmFit(dge, design)
lrt <- glmLRT(fit, contrast = c(0, 1))
DEGs <- topTags(lrt, n = Inf)$table
head(DEGs)
write.table(DEGs, "EdgeR_FC.txt", sep = "\t", quote = FALSE,
col.names = TRUE)
```



```
                 logFC   logCPM       LR       PValue          FDR
LOC102406514 -14.14468 6.964349 297.4628 1.176386e-66 1.877983e-62
DKK4         -14.56898 7.387691 291.2707 2.628390e-65 2.097981e-61
AKR1D1       -13.73296 6.553869 282.7053 1.932295e-63 1.028239e-59
LOC102389737 -14.74105 7.559454 282.0110 2.737590e-63 1.092572e-59
UPK3BL2       13.34490 8.623037 280.2687 6.561984e-63 2.095110e-59
LOC102415899  13.26465 6.082747 279.3495 1.040782e-62 2.769173e-59
```

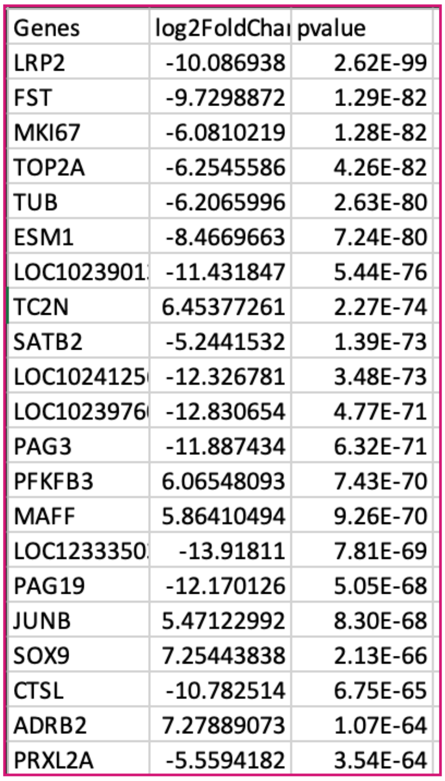
Note : padj and FDR are considered respectively from DESeq2 and EdgeR. - 6157 genes below 0.05 padj and 6221 genes below FDR



The list of 5864 genes were taken for further analysis - 4729 left after removing the LOC Ids - 1409 genes are positively regulated $\log_2$FC > 2.0 and 1203 genes are -vely regulated with $\log_2$FC > -2.0

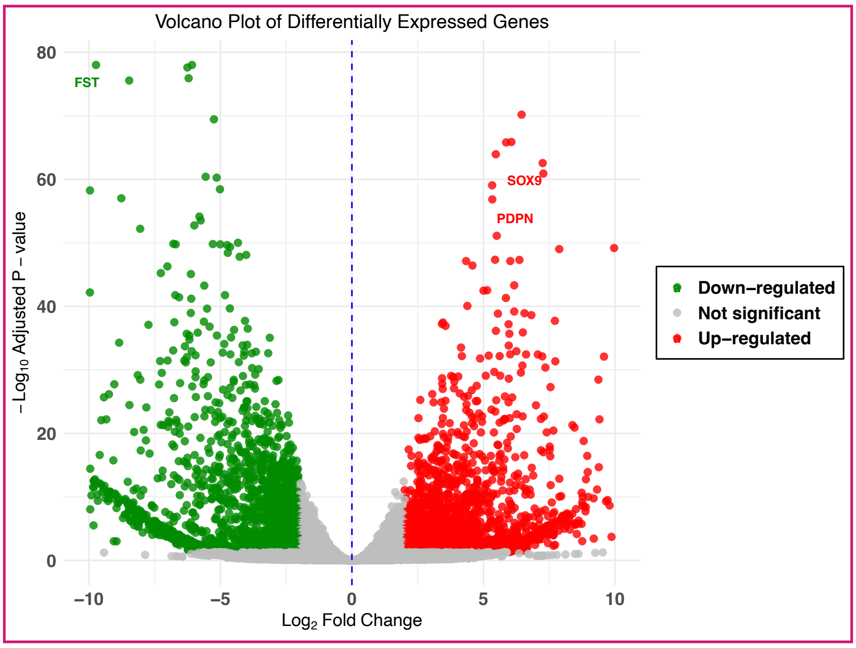
7. Valcano plot of the DE genes

Input file format : For_valcanofn.txt



Genes	log2FoldCha	pvalue
LRP2	-10.086938	2.62E-99
FST	-9.7298872	1.29E-82
MKI67	-6.0810219	1.28E-82
TOP2A	-6.2545586	4.26E-82
TUB	-6.2065996	2.63E-80
ESM1	-8.4669663	7.24E-80
LOC1023901	-11.431847	5.44E-76
TC2N	6.45377261	2.27E-74
SATB2	-5.2441532	1.39E-73
LOC1024125	-12.326781	3.48E-73
LOC1023976	-12.830654	4.77E-71
PAG3	-11.887434	6.32E-71
PFKFB3	6.06548093	7.43E-70
MAFF	5.86410494	9.26E-70
LOC1233350	-13.91811	7.81E-69
PAG19	-12.170126	5.05E-68
JUNB	5.47122992	8.30E-68
SOX9	7.25443838	2.13E-66
CTSL	-10.782514	6.75E-65
ADRB2	7.27889073	1.07E-64
PRXL2A	-5.5594182	3.54E-64

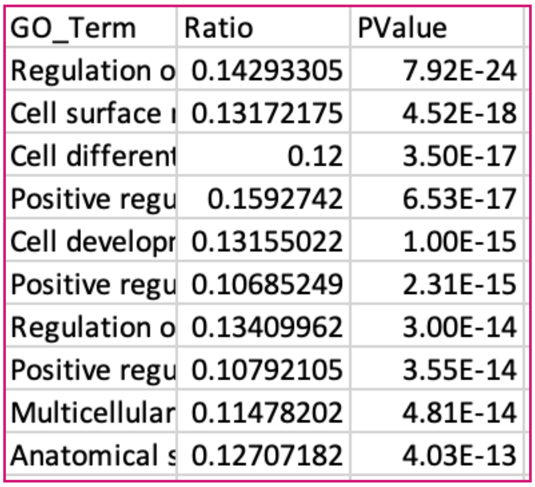
```
library(ggplot2)
library(dplyr)
library(readr)
library(ggrepel)
df <- read_delim("For_valcanofn.txt", delim = "\t")
df <- df %>%
  mutate(padj = p.adjust(pvalue, method = "BH"))
df <- df %>%
  filter(!is.na(log2FoldChange) & !is.na(padj))
log2FoldChange_limit <- 10
padj_limit <- 1e-300
df <- df %>%
  filter(abs(log2FoldChange) <= log2FoldChange_limit & padj >=
padj_limit)
df <- df %>%
  mutate(
    regulation = case_when(
      padj < 0.05 & log2FoldChange > 2 ~ "Up-regulated",
      padj < 0.05 & log2FoldChange < -2 ~ "Down-regulated",
      TRUE ~ "Not significant"
    )
  )
genes_to_annotate <- c("SOX9", "FST", "PDPN")
volcano_plot <- ggplot(df, aes(x = log2FoldChange, y =
-log10(padj), color = regulation)) +
  geom_point(alpha = 0.8, size = 2) +
  scale_color_manual(values = c("Not significant" = "grey", "Up-
regulated" = "red", "Down-regulated" = "green4")) +
  theme_minimal() +
  labs(
    title = "Volcano Plot of Differentially Expressed Genes",
    x = expression(Log[2]~Fold~Change),
    y = expression(-Log[10]~Adjusted~P-value)
  ) +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.title = element_blank(),
    legend.position = "right",
    legend.background = element_rect(color = "black", size = 0.5,
linetype = "solid"),
    legend.text = element_text(face = "bold", size = 12),
    axis.title.x = element_text(face = "bold", size = 12),
    axis.title.y = element_text(face = "bold", size = 12),
    axis.text = element_text(size = 12,face = "bold")
  ) +
  geom_vline(xintercept = 0, linetype = "dashed", color = "blue")
+
geom_text_repel(
  data = df %>% filter(Genes %in% genes_to_annotate),
  aes(label = Genes),
  box.padding = 0.5,
  point.padding = 0.5,
  segment.color = 'grey50',
  max.overlaps = Inf,
  fontface = "bold",
  size = 3
)
print(volcano_plot)
ggsave("volcano_plot.pdf", plot = volcano_plot, width = 8, height
= 6)
```



8. Functional annotation of the DE genes

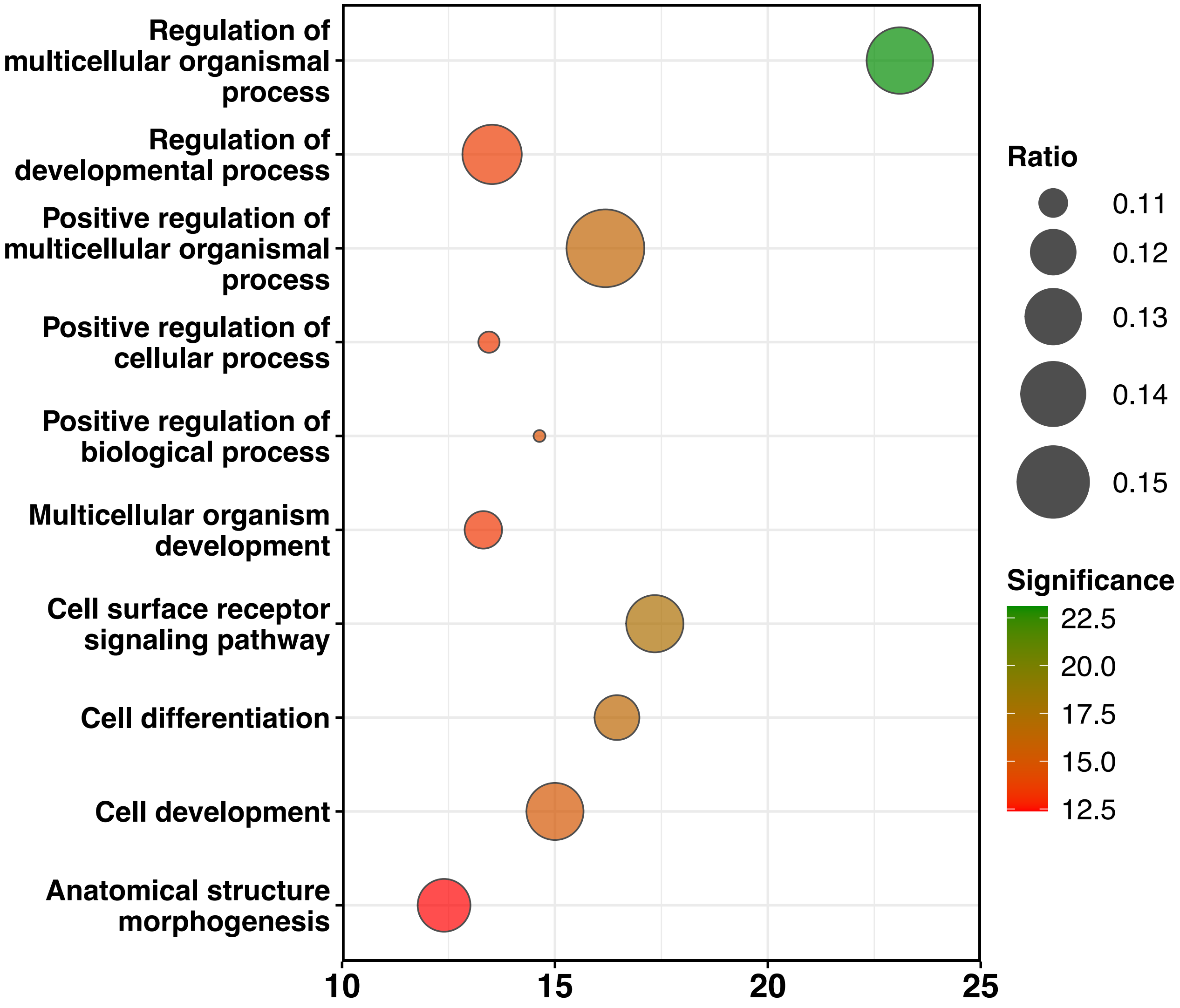
Functional annotation can be done in any of the available tools. Here it was done by Cluego the

Input file format: fa.txt



GO_Term	Ratio	PValue
Regulation o	0.14293305	7.92E-24
Cell surface ı	0.13172175	4.52E-18
Cell different	0.12	3.50E-17
Positive regu	0.1592742	6.53E-17
Cell developr	0.13155022	1.00E-15
Positive regu	0.10685249	2.31E-15
Regulation o	0.13409962	3.00E-14
Positive regu	0.10792105	3.55E-14
Multicellular	0.11478202	4.81E-14
Anatomical s	0.12707182	4.03E-13

```
library(ggplot2)
library(stringr)
data <- read.delim("fa.txt", header = TRUE, sep = "\t")
gg <- ggplot(data, aes(x = -log10(PValue), y = GO_Term, size =
Ratio, fill = -log10(PValue))) +
  geom_point(shape = 21, alpha = 0.7) +
  #labs(title = "GO Term Enrichment", x = "Enrichment", y = "GO
Term") +
  scale_size_continuous(range = c(2, 12), guide =
guide_legend(override.aes = list(fill = "black"))) +
  scale_fill_gradient(low = "red", high = "green4", name =
"Significance") +
  scale_y_discrete(labels = function(y) str_wrap(y, width = 25)) +
  theme_minimal() +
  theme(legend.text = element_text(size = 12), legend.title =
element_text(size = 12, face = "bold"),
        axis.text.x = element_text(size = 14, face = "bold",colour
= "Black"), axis.text.y = element_text(size = 12, face =
"bold",colour = "Black"),
        legend.margin = margin(b = 5),panel.border =
element_rect(color = "black", fill = NA, size = 1.0),axis.ticks =
element_line(),axis.title.x=element_blank(),axis.title.y=element_b
lank())
gg
ggsave("G_Inf_CD4_bubble.pdf", gg, width = 9, height = 10, units =
"in", bg = "white")
```


Regulation of multicellular organismal process
Regulation of developmental process
Positive regulation of multicellular organismal process
Positive regulation of cellular process
Positive regulation of biological process
Multicellular organism development
Cell surface receptor signaling pathway
Cell differentiation
Cell development
Anatomical structure morphogenesis
10
15
20
25
Ratio
0.11
0.12
0.13
0.14
0.15
Significance
22.5
20.0
17.5
15.0
12.5