RNA seq data analysis

Fastq

cat conA_rep1.fq.gz | head -n 20 (not a zip file)
zcat conA_rep1.fq.gz | head -n 20

```
ibab@IBAB-MSc14-Comp003:~/data_analysis$ cat conA_rep1.fq.gz | head -n 20
@A01811:33:HWGMLDSX5:4:2314:23059:3098 1:N:0:GGTGATGA+AGGCTATA
CCGTCGACACCCTCCTTGAACACCAAGAGGTTAACTAACTCGTCCAACTTGAAGGCCCCTTCCTTCCCAGCAGCAACAACTTGCTTCACCAGAC
GAGGAGA
+
FFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFF
@A01811:33:HWGMLDSX5:4:2457:26377:4366 1:N:0:GGTGATGA+AGGCTATA
GATGGTAGCAGCACCTGGAGTCCAGTTACATGGCAAGACAGTACCGTTCTTGTCGGTCCATTGGAAGGCTTCAACCAATCTCAAGGCTTCGTCA
ACGTTTC
+
,:,FFFFF:F:::FFFFFFFFF:FF:FFFFFFF:FFFFFF:FFFFFFFFFF:,,FF:FFF:FFF:FFF,FFFF::FFFFFFF:FFFFF,FFFFF
:F:FFFF
@A01811:33:HWGMLDSX5:3:2278:1298:9377 1:N:0:GGTGATGA+AGGCTATA
GTGGAAGTCTGTAAACTTGTGGAAGGAAAACTGGATTTAGTTCACATGTCAAATTAGAGAAAAAATTGGTATGGAGATCTTCAACAGGGCACGT
CAAGTAA
+
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFF
@A01811:33:HWGMLDSX5:4:2575:7093:26365 1:N:0:GGTGATGA+AGGCTATA
CCACTGAACTATTATTGATATTGCTGTCGGATATTAAATTCAAGCATTTGCTTAAAACGTTTACAATCTTGCTGGAGTTTTGGTTGAAGTTCAA
TCTCTGA
+
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFF
@A01811:33:HWGMLDSX5:2:2347:24379:35070 1:N:0:GGTGATGA+AGGCTATA
GGAGATCAAAATTCTGGAAGAACCACTTCTGAATTCCTTCATGATGGTGTCTCTTTCTTGTTGTGGTAAATCAGAGTAGATGGCAGAAACGGTA
AATTTGT
```

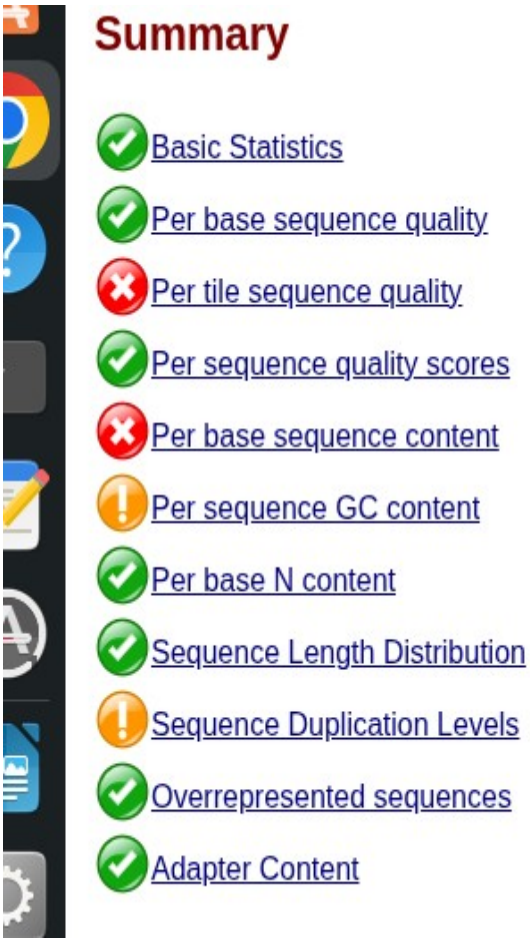
Running Fastqc

fastqc -o fastqc conA_rep2.fq

```
Skipping 'conA_rep1.fq' which didn't exist, or couldn't be read
(fastqc) ibab@IBAB-MSc14-Comp003:~/data_analysis$ fastqc -o fastqc conA_rep1.fq
null
Started analysis of conA_rep1.fq
Approx 5% complete for conA_rep1.fq
Approx 10% complete for conA_rep1.fq
Approx 15% complete for conA_rep1.fq
Approx 20% complete for conA_rep1.fq
Approx 25% complete for conA_rep1.fq
Approx 30% complete for conA_rep1.fq
Approx 35% complete for conA_rep1.fq
Approx 40% complete for conA_rep1.fq
Approx 45% complete for conA_rep1.fq
Approx 50% complete for conA_rep1.fq
Approx 55% complete for conA_rep1.fq
Approx 60% complete for conA_rep1.fq
Approx 65% complete for conA_rep1.fq
Approx 70% complete for conA_rep1.fq
Approx 75% complete for conA_rep1.fq
Approx 80% complete for conA_rep1.fq
Approx 85% complete for conA_rep1.fq
Approx 90% complete for conA_rep1.fq
Approx 95% complete for conA_rep1.fq
Approx 100% complete for conA_rep1.fq
Analysis complete for conA_rep1.fq
(fastqc) ibab@IBAB-MSc14-Comp003:~/data_analysis$ fastqc -o fastqc conA_rep2.fq
null
Started analysis of conA_rep2.fq
```

Fastqc results

```
conA_rep1_fastqc.html  conA_rep2_fastqc.html  conB_rep1_fastqc.html  conB_rep2_fastqc.html
conA_rep1_fastqc.zip   conA_rep2_fastqc.zip   conB_rep1_fastqc.zip   conB_rep2_fastqc.zip
(fastqc) ibab@IBAB-MSc14-Comp003:~/data_analysis$
```



Alignment to reference genome (using bowtie2)

Making index of referance genome

bowtie2-build CopyofGCF_000146045.2_R64_genomic.fna reference_index

Aligning reads to reference genome

bowtie2 -x reference_index -U "Copy of conA_rep1.fq.gz" -S conA.sam
bowtie2 -x reference_index -U "Copy of conA_rep1.fq.gz" -S conA1.sam
bowtie2 -x reference_index -U "Copy of conA_rep1.fq.gz" -S conB1.sam
bowtie2 -x reference_index -U "Copy of conA_rep1.fq.gz" -S conB2.sam

Viewing samfile(using samtools)

samtools view conA.sam | head

```
A01811:33:HWGMLDSX5:4:2314:23059:3098   0       NC_001143.9     524779  42      101M    *    0
0       CCGTCGACACCCTCCTTGAACACCAAGAGGTTAACTAACTCGTCCAACTTGAAGGCCCCTTCCTTCCCAGCAGCAACAACTTGCTT
CACCAGACGAGGAGA FFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFF AS:i:0  XN:i:0  XM:i:0  XO:i:0  XG:i:0  NM:i:0  MD:Z:101        YT:Z:U
U
A01811:33:HWGMLDSX5:4:2457:26377:4366   16      NC_001145.3     220577  42      101M    *    0
0       GAAACGTTGACGAAGCCTTGAGATTGGTTGAAGCCTTCCAATGGACCGACAAGAACGGTACTGTCTTGCCATGTAACTGGACTCCA
GGTGCTGCTACCATC FFFF:F:FFFFF,FFFFF:FFFFFFF::FFFF,FFF:FFF:FFF:FF,,:FFFFFFFFFF:FFFFFF:FFFFFFF:FF
:FFFFFFFFF:::F:FFFFF,:, AS:i:0  XN:i:0  XM:i:0  XO:i:0  XG:i:0  NM:i:0  MD:Z:101        YT:Z:U
U
A01811:33:HWGMLDSX5:3:2278:1298:9377    0       NC_001134.8     296809  42      101M    *    0
0       GTGGAAGTCTGTAAACTTGTGGAAGGAAAACTGGATTTAGTTCACATGTCAAATTAGAGAAAAAATTGGTATGGAGATCTTCAACA
GGGCACGTCAAGTAA FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFF AS:i:-5 XN:i:0  XM:i:1  XO:i:0  XG:i:0  NM:i:1  MD:Z:34G66      YT:Z:U
U
A01811:33:HWGMLDSX5:4:2575:7093:26365   0       NC_001142.9     92259   42      101M    *    0
0       CCACTGAACTATTATTGATATTGCTGTCGGATATTAAATTCAAGCATTTGCTTAAAACGTTTACAATCTTGCTGGAGTTTTGGTTG
AAGTTCAATCTCTGA FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFF AS:i:0  XN:i:0  XM:i:0  XO:i:0  XG:i:0  NM:i:0  MD:Z:101        YT:Z:U
U
A01811:33:HWGMLDSX5:2:2347:24379:35070  16      NC_001143.9     555834  6       101M    *    0
0       ACAAATTTACCGTTTCTGCCATCTACTCTGATTTACCACAACAAGAAAGAGACACCATCATGAAGGAATTCAGAAGTGGTTCTTCC
AGAATTTTGATCTCC FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFF AS:i:0  XS:i:-5 XN:i:0  XM:i:0  XO:i:0  XG:i:0  NM:i:0  MD:Z:101     Y
T:Z:UU
A01811:33:HWGMLDSX5:4:2635:7030:6183    0       NC_001134.8     162957  42      101M    *    0
0       CACTCATTAAACACCAAAAATCCAAAGAAGAAAAAAAGGAAAATGTGAGAAAGAATTTAGATTAAGAATCAAGCCAGATTAGACTT
ATTTGAACTTCTTAC FFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFF:FF::FFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFF,FFFFFFFFFFFFFFFF AS:i:0  XN:i:0  XM:i:0  XO:i:0  XG:i:0  NM:i:0  MD:Z:101        YT:Z:U
U
```

Extracting reads count for each gene

```
library(Rsubread)
# Run featureCounts on all 4 SAM files together
out_FeatCount <- featureCounts(
  files = c("conA.sam", "conA1.sam", "conB1.sam", "conB2.sam"),
  annot.ext = "Copy of GCF_000146045.2_R64_genomic.gtf",
  isGTFAnnotationFile = TRUE,
  GTF.featureType = "exon",
  GTF.attrType = "gene_id",
  nthreads = 8  # adjust based on your CPU
)

# Extract counts matrix
counts_matrix <- as.data.frame(out_FeatCount$counts)

print(head(counts_matrix))
```

```
         conA.sam conA1.sam conB1.sam conB2.sam
YAL068C         0         0         0         0
YAL067W-A       0         0         0         0
YAL067C         0         0         0         0
YAL065C         0         0         0         0
YAL064W-B       1         0         1         0
YAL064C-A       0         0         0         0
```

Differential gene expression using Deseq2

```
# Data import
counts <- read.table("all_samples_counts.csv",
                     header = TRUE,
                     row.names = 1,
                     sep = "\t",
                     check.names = FALSE)


suppressPackageStartupMessages(library(DESeq2)) # to load DESeq2 and suppress the long startup message

# Make metadata
coldata <- data.frame(
  row.names = colnames(counts),
  condition = c("control", "control", "treated", "treated")
)

# Convert condition to factor
coldata$condition <- factor(coldata$condition)

# Build DESeq2 dataset
dds <- DESeqDataSetFromMatrix(countData = counts,
                              colData = coldata,
                              design = ~ condition)
print(dds)
```

```
class: DESeqDataSet
dim: 6459 4
metadata(1): version
assays(1): counts
rownames(6459): YAL068C YAL067W-A ... tM(CAU)Q2 Q0285
rowData names(0):
colnames(4): conA conA1 conB1 conB2
colData names(1): condition
>
```

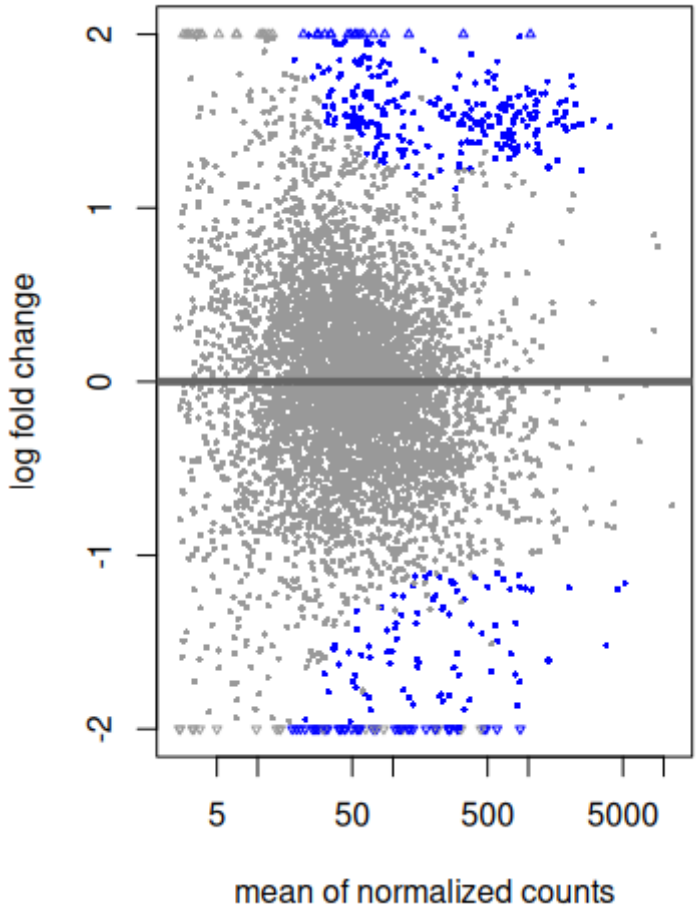
```
 4. Pre-filter (optional, removes low counts)
dds <- dds[rowSums(counts(dds)) > 10, ]

# 5. Run DESeq2
dds <- DESeq(dds)

# 6. Get results (e.g., condition treated vs control)
res <- results(dds, contrast = c("condition","treated","control"))
```

 # 7. Order by adjusted p-value
resOrdered <- res[order(res$padj),]

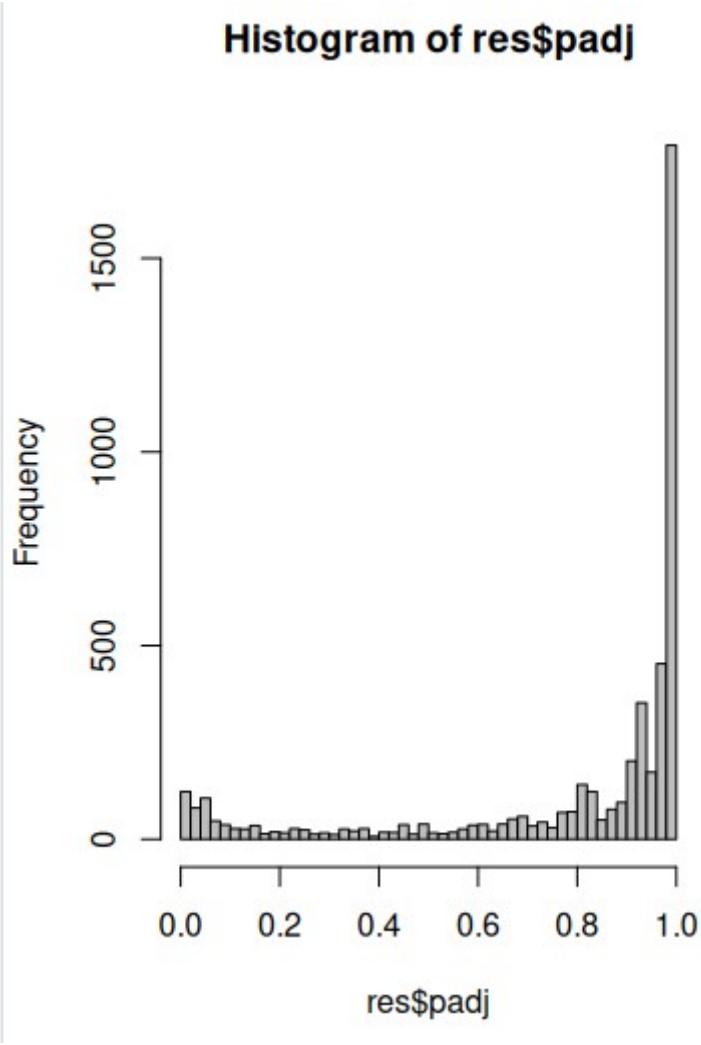
 # 9. QC plots
plotMA(res, ylim=c(-2,2)) # MA plot
hist(res$padj, breaks=50, col="grey") # p-value distribution



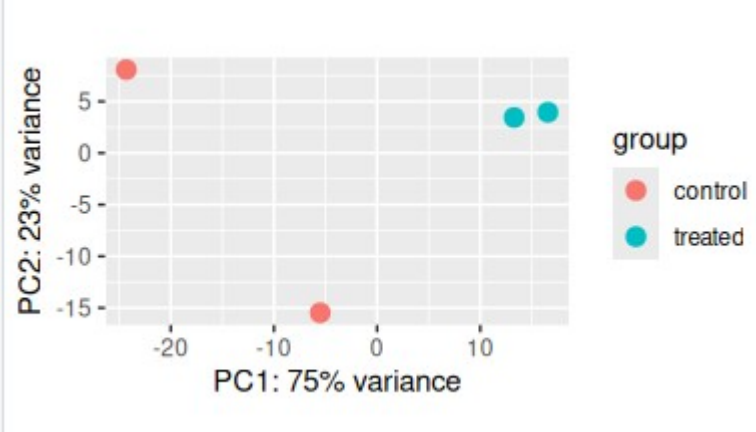
Interpretation

- Genes with **high log fold change** (far from zero on the y-axis, either positive or negative) are likely to be up-regulated or down-regulated between the two conditions.
- Genes with **low mean expression** tend to have more scattered log fold changes, which can reflect higher variability and less reliable differential expression at very low counts.
- The **majority of genes cluster around log fold change = 0**, meaning their expression doesn't change between conditions.
- **Blue points at the top and bottom**: These represent genes that are strongly up-regulated (top blue) or down-regulated (bottom blue) and are statistically significant.

- The **horizontal line at 0** on the y-axis is the reference for no change in expression.

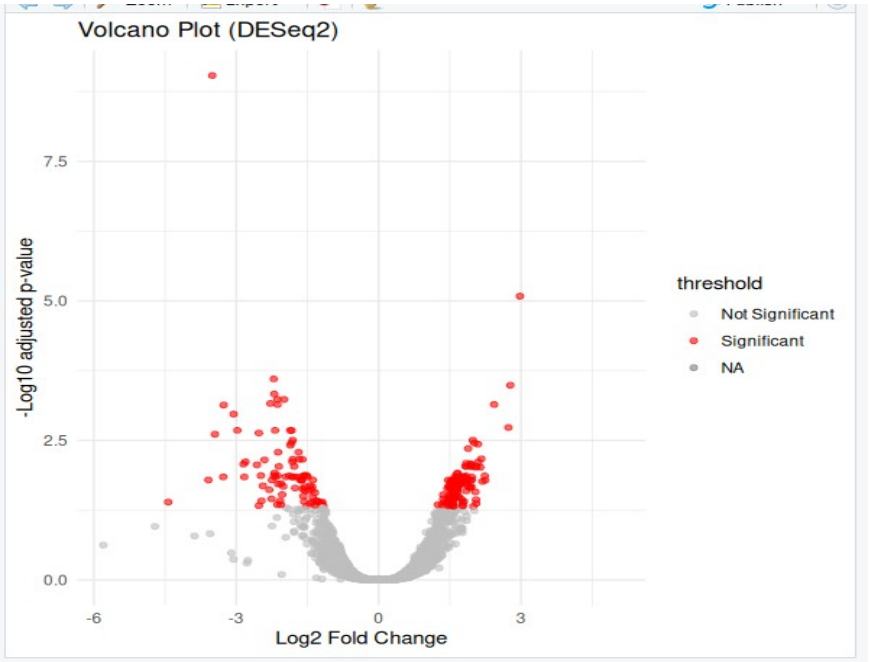


- The vast majority of genes are not significant after adjustment for multiple comparisons, as shown
- by the high frequency at a padj of 1.
- Genes with padj near zero (left side of plot) are candidates for significant differential expression.
- This distribution is typical in RNA-seq experiments, where only a small proportion of genes pass the significance threshold.
- The histogram helps validate that multiple testing correction was performed and that significance was not over-called.



Interpretation:

- Samples from the same group cluster close together, indicating similar overall gene expression profiles.
- Control samples are distinct and separated from treated samples along PC1, suggesting global transcriptional differences between the two groups.
- PC1 captures most of the variation, which corresponds to the experimental condition.



Interpretation

- **Significant genes** (red):
  - Located at the top left (down-regulated) and top right (up-regulated) of the plot.
  - These genes show both strong fold-change and strong statistical support.
- **Non-significant genes** (gray):
  - Clustered near zero-fold change or have low -log10 p-values.

        - These genes do not show robust evidence for differential expression.
- The plot highlights which genes are both highly differentially expressed and statistically significant.