Class17

Nancy Leon-Rivera PID A12925462

Downstream analysis

```
folders <- list.files(pattern = "_quant")
files <- paste0(folders, "/abundance.h5")
```

```
file.exists(files)
```

```
[1] TRUE TRUE TRUE TRUE
```

Can also use the following to check if the files exist:

```
#BiocManager::install("tximport")
# Load the package
library(rhdf5)
library(tximport)

# setup the folder and filenames to read
folders <- list.files(pattern = "_quant")
files <- file.path(folders, "/abundance.h5")

txi.kallisto <- tximport(files, type = "kallisto", txOut = TRUE)
```

```
1 2 3 4
```

```
names(files) <- sub("_quant", "", folders)
files
```

```
                      SRR2156848                       SRR2156849
"SRR2156848_quant//abundance.h5" "SRR2156849_quant//abundance.h5"
                      SRR2156850                       SRR2156851
"SRR2156850_quant//abundance.h5" "SRR2156851_quant//abundance.h5"
```

Load up the tximport library

```
library(tximport)

txi.kallisto <- tximport(files, type = "kallisto", txOut = T)
```

```
1 2 3 4
```

```
head(files)
```

```
                      SRR2156848                       SRR2156849
"SRR2156848_quant//abundance.h5" "SRR2156849_quant//abundance.h5"
                      SRR2156850                       SRR2156851
"SRR2156850_quant//abundance.h5" "SRR2156851_quant//abundance.h5"
```

```
head(txi.kallisto$counts)
```

```
                SRR2156848 SRR2156849 SRR2156850 SRR2156851
ENST00000539570          0          0    0.00000          0
ENST00000576455          0          0    2.62037          0
ENST00000510508          0          0    0.00000          0
ENST00000474471          0          1    1.00000          0
ENST00000381700          0          0    0.00000          0
ENST00000445946          0          0    0.00000          0
```

Remove zero count genes

Before subsequent analysis, we might want to filter out those annotated transcripts with no reads:

```
to.keep <- rowSums(txi.kallisto$counts) > 0
kset.nonzero <- txi.kallisto$counts[to.keep,]
```

```
nrow(kset.nonzero)
```

```
[1] 94561
```

Or can use:

```
sum(rowSums(txi.kallisto$counts)>0)
```

```
[1] 94561
```

```
colSums(txi.kallisto$counts)
```

```
SRR2156848 SRR2156849 SRR2156850 SRR2156851
   2563611    2600800    2372309    2111474
```

And those with no change over the samples:

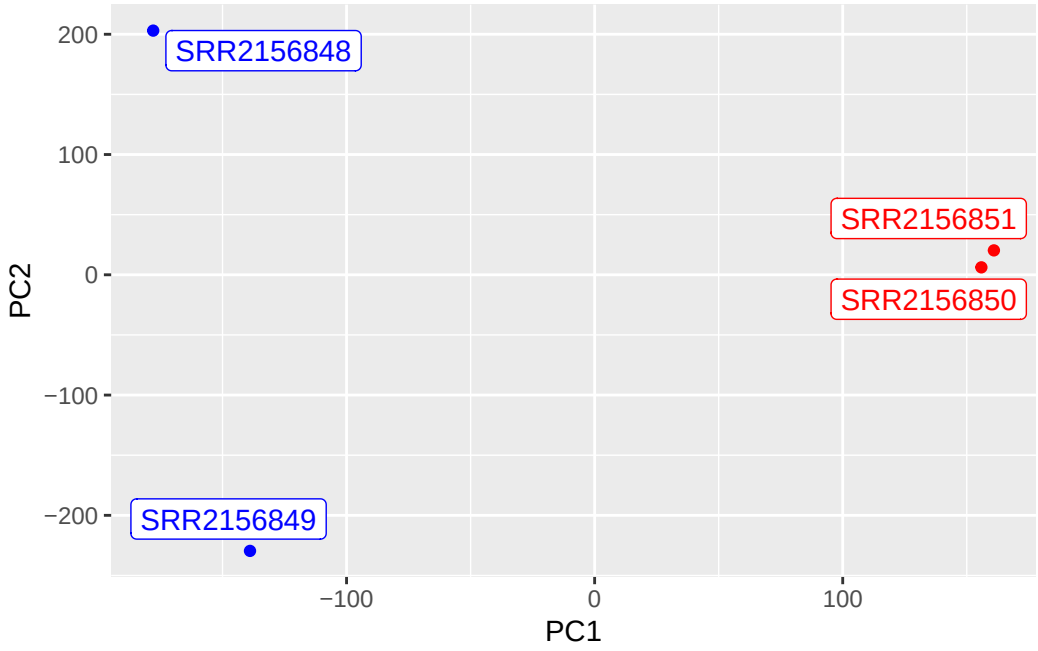
```
keep2 <- apply(kset.nonzero,1,sd)>0
x <- kset.nonzero[keep2,]
```

Try a PCA

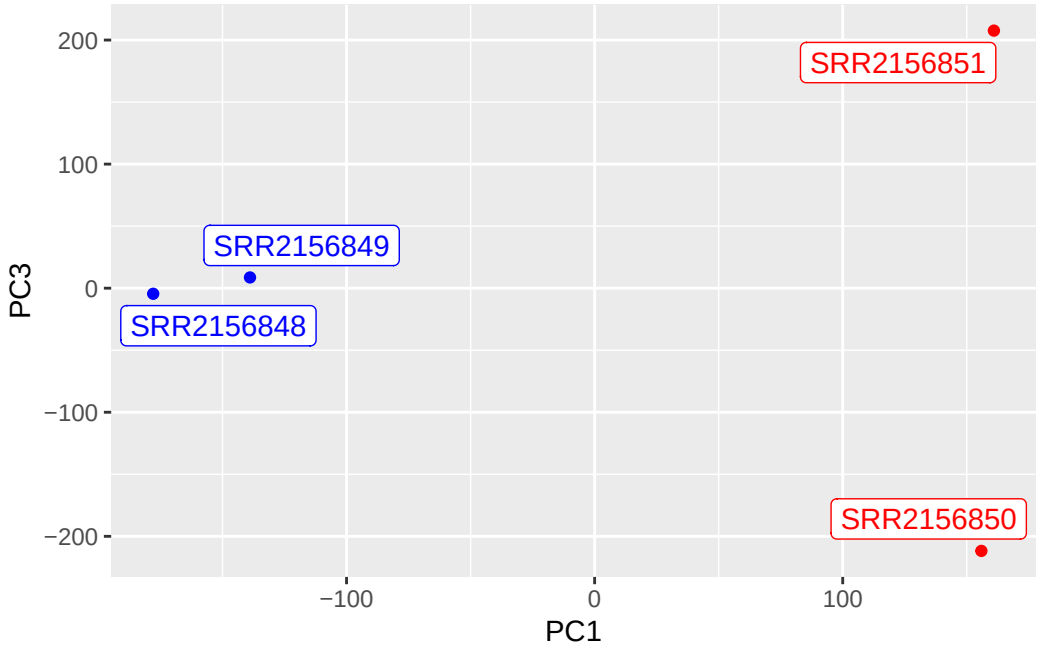
```
pca <- prcomp(t(x), scale=T)
summary(pca)
```

```
Importance of components:
                            PC1      PC2      PC3   PC4
Standard deviation     183.6379 177.3605 171.3020 1e+00
Proportion of Variance   0.3568   0.3328   0.3104 1e-05
Cumulative Proportion    0.3568   0.6895   1.0000 1e+00
```

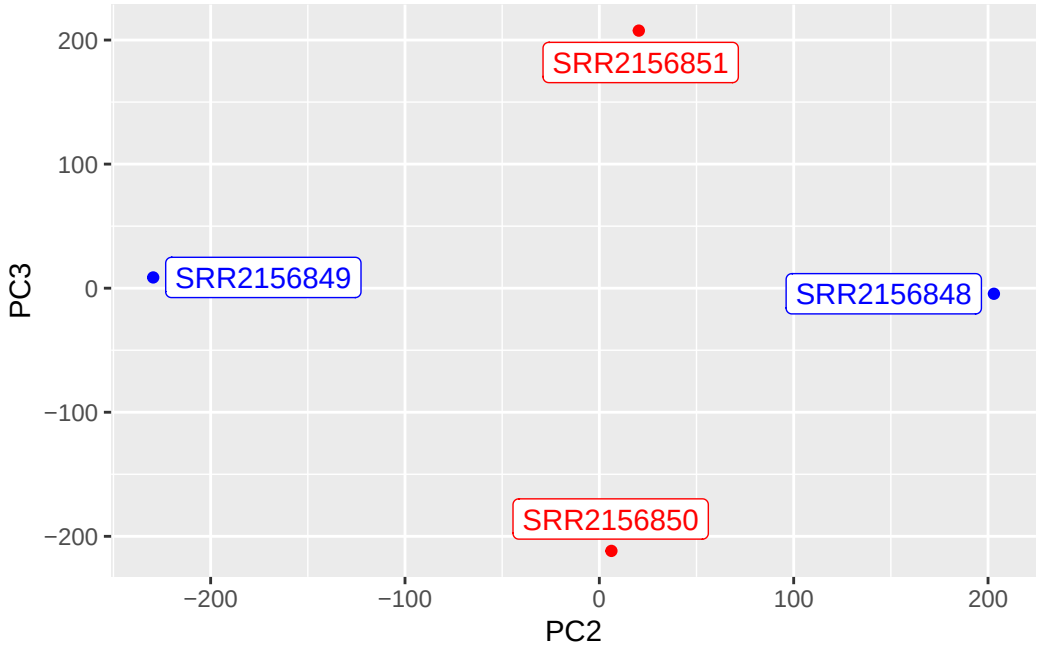
```
library(ggplot2)
library(ggrepel)
mycols <- c("blue","blue", "red", "red")
ggplot(pca$x)+
         aes(PC1, PC2)+
         geom_point(col=mycols)+
  geom_label_repel(label=rownames(pca$x), col=mycols)
```



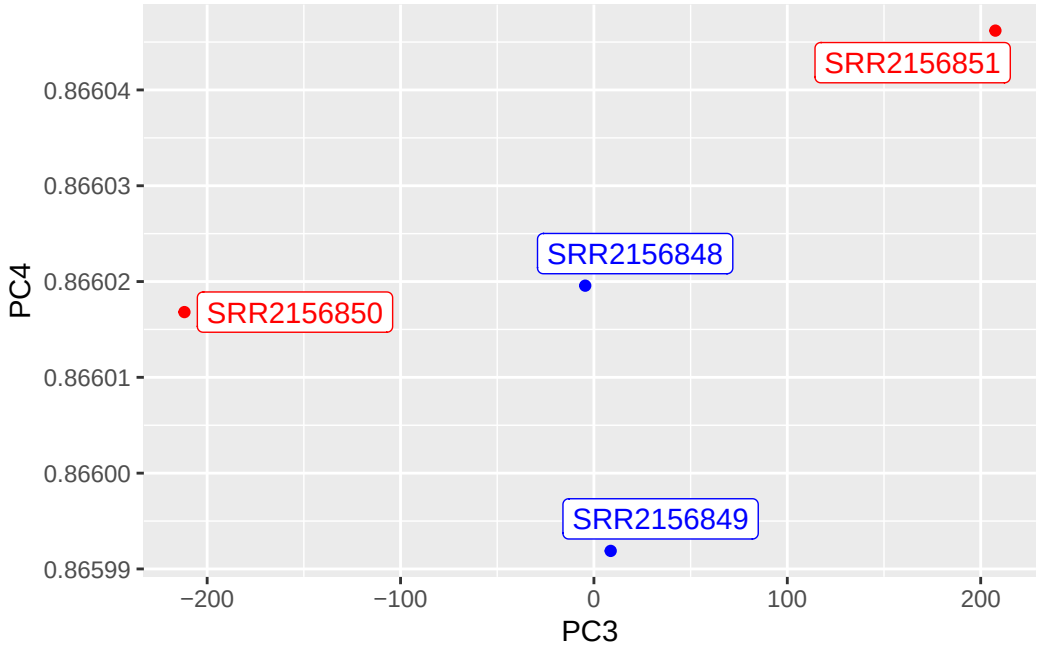

```
mycols <- c("blue","blue", "red", "red")
ggplot(pca$x)+
         aes(PC1, PC3)+
         geom_point(col=mycols)+
  geom_label_repel(label=rownames(pca$x), col=mycols)
```




```
mycols <- c("blue","blue", "red", "red")
ggplot(pca$x)+
         aes(PC2, PC3)+
         geom_point(col=mycols)+
  geom_label_repel(label=rownames(pca$x), col=mycols)
```




```
mycols <- c("blue","blue", "red", "red")
ggplot(pca$x)+
         aes(PC3, PC4)+
         geom_point(col=mycols)+
  geom_label_repel(label=rownames(pca$x), col=mycols)
```



DESeq

```
library(DESeq2)
```

```
Loading required package: S4Vectors
```

```
Loading required package: stats4
```

```
Loading required package: BiocGenerics
```

```

Attaching package: 'BiocGenerics'
```

```
The following objects are masked from 'package:stats':

    IQR, mad, sd, var, xtabs
```

```
The following objects are masked from 'package:base':

    anyDuplicated, aperm, append, as.data.frame, basename, cbind,
    colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
    get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
    match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
    Position, rank, rbind, Reduce, rownames, sapply, saveRDS, setdiff,
    table, tapply, union, unique, unsplit, which.max, which.min


Attaching package: 'S4Vectors'

The following object is masked from 'package:utils':

    findMatches

The following objects are masked from 'package:base':

    expand.grid, I, unname

Loading required package: IRanges


Attaching package: 'IRanges'

The following object is masked from 'package:grDevices':

    windows

Loading required package: GenomicRanges

Loading required package: GenomeInfoDb

Loading required package: SummarizedExperiment

Loading required package: MatrixGenerics

Loading required package: matrixStats
```

```
Attaching package: 'MatrixGenerics'

The following objects are masked from 'package:matrixStats':

    colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
    colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
    colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
    colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
    colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
    colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
    colWeightedMeans, colWeightedMedians, colWeightedSds,
    colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
    rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
    rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
    rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
    rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
    rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
    rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
    rowWeightedSds, rowWeightedVars

Loading required package: Biobase

Welcome to Bioconductor

    Vignettes contain introductory material; view with
    'browseVignettes()'. To cite Bioconductor, see
    'citation("Biobase")', and for packages 'citation("pkgname")'.


Attaching package: 'Biobase'

The following object is masked from 'package:MatrixGenerics':

    rowMedians

The following objects are masked from 'package:matrixStats':

    anyMissing, rowMedians
```

```
sampleTable <- data.frame(condition = factor(rep(c("control", "treatment"), each = 2)))
rownames(sampleTable) <- colnames(txi.kallisto$counts)

sampleTable
```

```
           condition
SRR2156848   control
SRR2156849   control
SRR2156850 treatment
SRR2156851 treatment
```

```
dds <- DESeqDataSetFromTximport(txi.kallisto,
                                sampleTable,
                                ~condition)
```

```
using counts and average transcript lengths from tximport
```

```
dds <- DESeq(dds)
```

```
estimating size factors

using 'avgTxLength' from assays(dds), correcting for library size

estimating dispersions

gene-wise dispersion estimates

mean-dispersion relationship

-- note: fitType='parametric', but the dispersion trend was not well captured by the
   function: y = a/x + b, and a local regression fit was automatically substituted.
   specify fitType='local' or 'mean' to avoid this message next time.

final dispersion estimates

fitting model and testing
```

```
res <- results(dds)
res
```

```
log2 fold change (MLE): condition treatment vs control
Wald test p-value: condition treatment vs control
DataFrame with 176981 rows and 6 columns
                 baseMean log2FoldChange     lfcSE      stat    pvalue
                <numeric>      <numeric> <numeric> <numeric> <numeric>
ENST00000539570  0.000000             NA        NA        NA        NA
ENST00000576455  0.761453       3.155061   4.86052 0.6491203  0.516261
ENST00000510508  0.000000             NA        NA        NA        NA
ENST00000474471  0.484938       0.181923   4.24871 0.0428185  0.965846
ENST00000381700  0.000000             NA        NA        NA        NA
...                   ...            ...       ...       ...       ...
ENST00000570559   1.03492       0.269118   3.32664 0.0808979  0.935523
ENST00000576031   0.00000             NA        NA        NA        NA
ENST00000577049   0.00000             NA        NA        NA        NA
ENST00000577091   0.00000             NA        NA        NA        NA
ENST00000576929   0.00000             NA        NA        NA        NA
                    padj
                <numeric>
ENST00000539570        NA
ENST00000576455        NA
ENST00000510508        NA
ENST00000474471        NA
ENST00000381700        NA
...                   ...
ENST00000570559        NA
ENST00000576031        NA
ENST00000577049        NA
ENST00000577091        NA
ENST00000576929        NA
```