STAR(QuantMode) analysis

Jasveer Thethi

2026-03-02

The analysis of the STAR alignment output files, following the use of –quantMode, to analyse ReadsPerGene.out.tab output

```
# define directory path of
quant_dir2 = "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_fea

# search the directory for files ending in ReadsPerGene.out.tab.
files = list.files(
  quant_dir2,
  pattern = "ReadsPerGene.out.tab",
   recursive = FALSE,
  full.names = TRUE
)
files
```

```
##  [1] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
##  [2] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
##  [3] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
##  [4] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
##  [5] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
##  [6] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
##  [7] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
##  [8] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
##  [9] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
## [10] "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureC
```

```
# extract sample names
sample_names2 = gsub("_ReadsPerGene.out.tab$","", basename(files))
# uses basename() to strip away the long folder path and gsub() to remove the file extension suffix
sample_names2
```

```
##  [1] "SRR3474721" "SRR3474722" "SRR3474723" "SRR3474724" "SRR3474725"
##  [6] "SRR3474726" "SRR3474727" "SRR3474728" "SRR3474729" "SRR3474730"
```

```
# link names to file paths
names(files) = sample_names2
files
```

```
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
```

```
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
##
## "/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star/star_featureCounts
```

```
# Merge and Format Count Data
# Load data.table for fast file reading using the fread function
library(data.table)

# read the first file and selecting columns 1 (GeneID) and 4 (Counts)
countData = data.frame(fread(files[1]))[c(1,4)]

# Loop through the remaining files
for(i in 2:length(files)) {
   # Column-bind the 4th column (counts) from each subsequent file to the main table
    countData = cbind(countData, data.frame(fread(files[i]))[4])
}

# Remove the first 4 rows of the dataframe
countData = countData[c(5:nrow(countData)), ]

# Assign descriptive names to columns, using "GeneID" for the first and sample IDs for the rest
colnames(countData) = c("GeneID", sample_names2)

# Set the "GeneID" column as the official row names for easier indexing
rownames(countData) = countData$GeneID

# Remove the first column (the redundant GeneID column) to leave only numerical count data
countData = countData[, -1]
```

```
# Load the Metadata
meta_data2 = read.csv("/home/jkt21/mnt/network/bioinformatics/users/jkt21/rnaseq_GSE81089_workflow/star

# set row names
# assigns the values in column X (usually the Sample IDs like "SRR...") as the actual row names of the
rownames(meta_data2) = meta_data2$X

# clean up the table
# deletes the column named X from the data frame
meta_data2$X = NULL
```

```r
# synchronize counts and metadata
# reorders the columns of your countData matrix to match the exact order of the rows in meta_data2
countData = countData[, rownames(meta_data2)]

# do > all(colnames(countData) == rownames(meta_data2)) to check if the column names of CountData match
```

```r
library(DESeq2)
```

```
## Loading required package: S4Vectors

## Loading required package: stats4

## Loading required package: BiocGenerics

## Loading required package: generics

##
## Attaching package: 'generics'

## The following objects are masked from 'package:base':
##
##     as.difftime, as.factor, as.ordered, intersect, is.element, setdiff,
##     setequal, union

##
## 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, is.unsorted, lapply, Map, mapply, match, mget,
##     order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##     rbind, Reduce, rownames, sapply, saveRDS, table, tapply, unique,
##     unsplit, which.max, which.min

##
## Attaching package: 'S4Vectors'

## The following objects are masked from 'package:data.table':
##
##     first, second

## 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:data.table':
##
##     shift

## Loading required package: GenomicRanges

## Loading required package: Seqinfo

## 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
```

```
# create a DESeqDataSet object from a count matrix
# 'countData' contains raw gene counts
# 'colData' contains sample metadata (sex and smoking status)
# The design formula '~sex + smoking' tells DESeq2 to model gene expression
# based on smoking status while controlling for the effect of sex.
dds2 = DESeqDataSetFromMatrix(countData = countData,
colData = meta_data2, design =  ~sex + smoking)
```

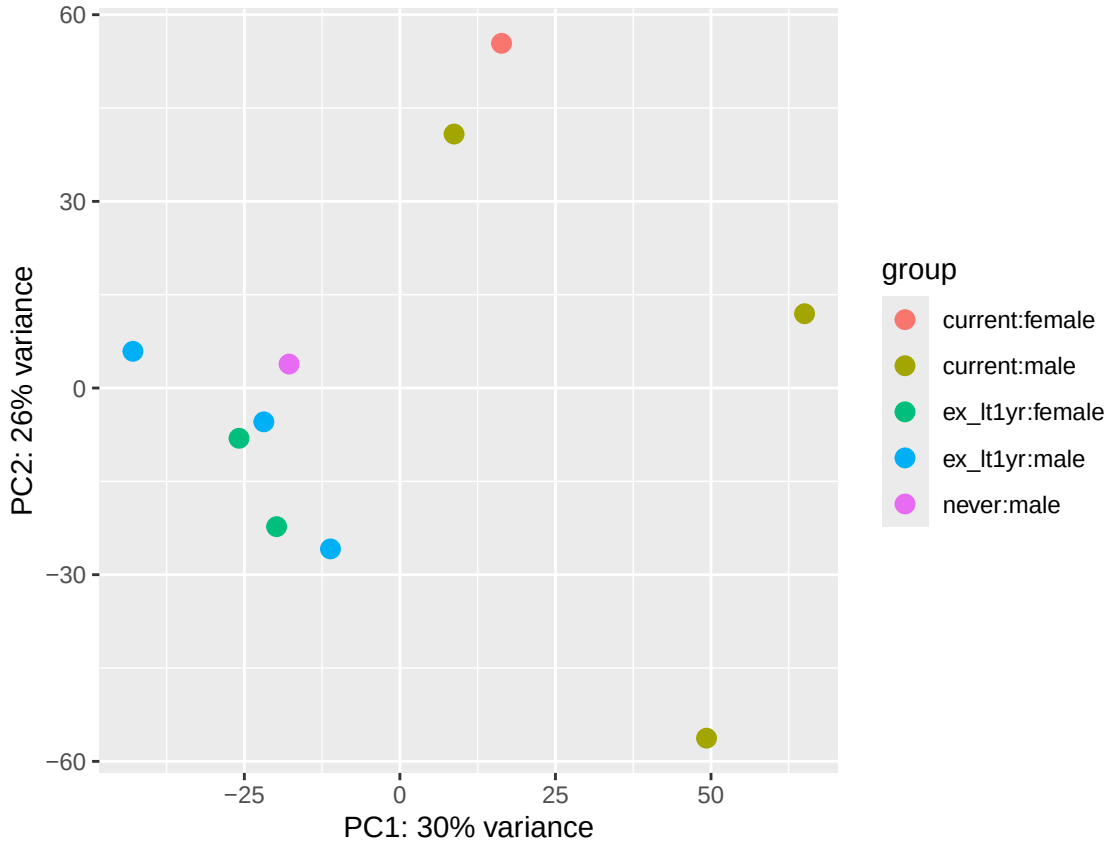
```
## Warning in DESeqDataSet(se, design = design, ignoreRank): some variables in
## design formula are characters, converting to factors
```

```
dds2 = DESeq(dds2)
```

```
## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing
```

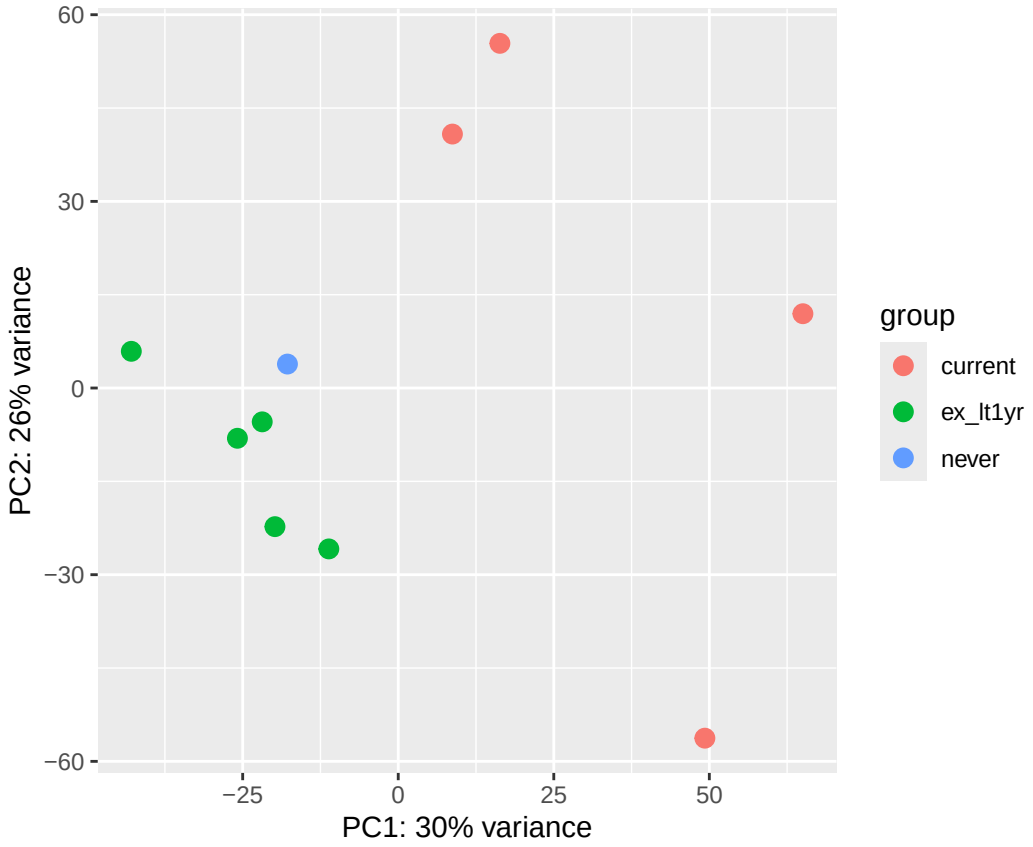
```
# Apply Variance Stabilising Transformation (VST)
vsdata2 = vst(dds2, blind = TRUE)

plotPCA(vsdata2, intgroup = c("smoking", "sex"))
```

```
## using ntop=500 top features by variance
```



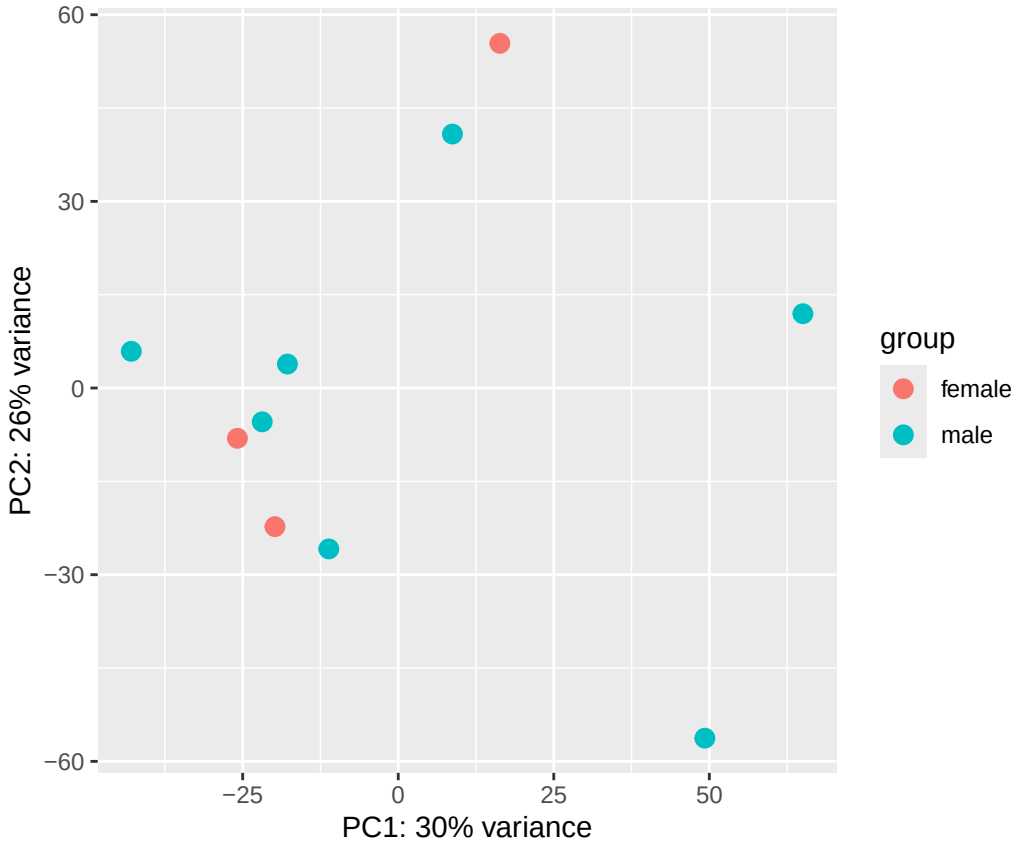

```
plotPCA(vsdata2, intgroup = "smoking")
```

```
## using ntop=500 top features by variance
```




```
plotPCA(vsdata2, intgroup = "sex")
```

```
## using ntop=500 top features by variance
```




```
# # Load the Independent Hypothesis Weighting package
# library(IHW)
#
# # Extracting results, comparing smokers and non-smokers
# res2 = results(
#   dds2,
#   contrast=c("smoking", "current", "never"),
#   alpha=0.05,
#   filterFun=IHW::ihw)
# res2
```

```
# summary(res2)
```

```
# plotMA(res2, alpha = 0.05)
```