|  |
| --- |
| > # The script can be used to statistically analyze differential expression of genes  > # and generate visualizations of the DEGs using outputs of Star mapped files  > # The inputs used here were pre-processed using bash scripts found in the main directory  >  > # The following script was modified from https://biocorecrg.github.io/RNAseq\_course\_2019/differential\_expression.html  >  >  > # Load the DESeq2 package  >  > library(DESeq2)  > library(tidyverse)  > # read in the sample sheet  > # header = TRUE: the first row is the "header", i.e. it contains the column names.  > # sep = "\t": the columns/fields are separated with tabs.  >  > # Copy the SampleSheet to the current directory  > sampletable <- read.table("sample\_sheet.txt", header=T, sep="\t")  >  > # add the sample names as row names (it is needed for some of the DESeq functions)  > rownames(sampletable) <- sampletable$SampleName  >  > # display the first 6 rows  > head(sampletable)  SampleName TISSUE  1 1 Heart  2 2 Heart  3 3 Heart  4 4 Heart  5 5 Heart  6 6 Liver  >  > # check the number of rows and the number of columns  > nrow(sampletable)  [1] 15  > ncol(sampletable)  [1] 2  >  > # create the DESeq object  > # countData is the matrix containing the counts  > # sampletable is the sample sheet / metadata we created  > # design is how we wish to model the data: what we want to measure here is the difference between the treatment times  > # Option 1 that reads in a matrix (we will not do it here):  >  > # first read in the matrix  > count\_matrix <- read.table("raw\_counts\_matrix.txt", header = T, row.names = 1)  > head(count\_matrix)  X1 X2 X3 X4 X5 X6 X7 X8 X9 X10 X11 X12 X13 X14 X15  LOC114595581 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  LOC114595943 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  LOC114596289 17 21 6 7 5 2 6 5 2 5 8 0 12 5 3  LOC114585404 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  LOC114597600 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  LOC114600179 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  > colnames(count\_matrix) <- sampletable$SampleName # The sample names are not showing in columns  > head(count\_matrix)  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15  LOC114595581 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  LOC114595943 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  LOC114596289 17 21 6 7 5 2 6 5 2 5 8 0 12 5 3  LOC114585404 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  LOC114597600 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  LOC114600179 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  > ncol(count\_matrix)  [1] 15  > nrow(count\_matrix)  [1] 24946  >  >  > # then create the DESeq object  > # countData is the matrix containing the counts  > # sampletable is the sample sheet / metadata we created  > # design is how we wish to model the data: what we want to measure here is the difference between the treatment times  >  > Deseq\_contrast\_data <- DESeqDataSetFromMatrix(countData = count\_matrix,  + colData = sampletable,  + design = ~TISSUE)  Warning message:  In DESeqDataSet(se, design = design, ignoreRank) :  some variables in design formula are characters, converting to factors  >  >  >  > # ----------------- F I L T E R I N G -------------------------  > # Number of genes before filtering:  > nrow(Deseq\_contrast\_data)  [1] 24946  >  > # Filter  > Deseq\_contrast\_data <- Deseq\_contrast\_data[rowSums(counts(Deseq\_contrast\_data)) > 0, ]  >  >  > # Number of genes left after low-count filtering:  > nrow(Deseq\_contrast\_data)  [1] 9707  >  > # test contrast  >  > Deseq\_con\_MLE <- DESeq(Deseq\_contrast\_data, modelMatrixType="expanded", betaPrior=T)  estimating size factors  estimating dispersions  gene-wise dispersion estimates  mean-dispersion relationship  final dispersion estimates  fitting model and testing  > Deseq\_con\_MLE  class: DESeqDataSet  dim: 9707 15  metadata(1): version  assays(4): counts mu H cooks  rownames(9707): LOC114596289 CWF19L1 ... LOC114590466 LOC114590505  rowData names(33): baseMean baseVar ... deviance maxCooks  colnames(15): 1 2 ... 14 15  colData names(3): SampleName TISSUE sizeFactor  >  >  > # Checking group names  > resultsNames(Deseq\_con\_MLE)  [1] "Intercept" "TISSUEHeart" "TISSUELiver" "TISSUESkeletal\_Muscle"  >  > # Create pairwise contrast matrix for all groups in the treatment column  >  > contrast1 = c("TISSUE", "Liver", "Skeletal\_Muscle")  > contrast2 = c("TISSUE", "Heart", "Skeletal\_Muscle")  > contrast3 = c("TISSUE", "Heart", "Liver")  >  > # running contrasts  > contrast1\_res\_MLE <- results(Deseq\_con\_MLE, contrast=contrast1, alpha = 0.05)  > contrast2\_res\_MLE <- results(Deseq\_con\_MLE, contrast=contrast2, alpha = 0.05)  > contrast3\_res\_MLE <- results(Deseq\_con\_MLE, contrast=contrast3, alpha = 0.05)  >  >  > # writing csv of all results  > write.csv(as.matrix(contrast1\_res\_MLE), file = "Liver\_Muscle\_DESeq2.csv", row.names = T)  > write.csv(as.matrix(contrast2\_res\_MLE), file = "Heart\_Muscle\_DESeq2.csv", row.names = T)  > write.csv(as.matrix(contrast3\_res\_MLE), file = "Heart\_Liver\_DESeq2.csv", row.names = T) |
|  |
| |  | | --- | | > | |