

# Mouse DGEs of SACC-PHHs mono-infected with HBV or coinfecte with HBV/HDV factors donor time

## Purpose:

To determine the DGE profiles (for mouse genes), relative to uninfected controls, of self-assembling co-cultures of primary human hepatocytes (SACC-PHHs) (co-cultured with 3T3J mouse non-parenchymal cells) mono-infected with HBV or co-infected with HBV/HDV at 8 and 28 days post-infection. Here, donor and time are factors in the design.

```
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(stringr)
library(ggplot2)
library(reshape2)
library(openxlsx)
library(DESeq2)

## Loading required package: S4Vectors
## Loading required package: stats4
## Loading required package: BiocGenerics
## Loading required package: parallel
##
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB

## The following objects are masked from 'package:dplyr':
##
##     combine, intersect, setdiff, union

## The following objects are masked from 'package:stats':
##
##     IQR, mad, xtabs

## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, cbind, colnames,
```

```

##      do.call, duplicated, eval, evalq, Filter, Find, get, grep,
##      grepl, intersect, is.unsorted, lapply, lengths, Map, mapply,
##      match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
##      Position, rank, rbind, Reduce, rownames, sapply, setdiff,
##      sort, table, tapply, union, unique, unsplit

##
## Attaching package: 'S4Vectors'

## The following objects are masked from 'package:dplyr':
##      first, rename

## The following objects are masked from 'package:base':
##      colMeans, colSums, expand.grid, rowMeans, rowSums

## Loading required package: IRanges

##
## Attaching package: 'IRanges'

## The following objects are masked from 'package:dplyr':
##      collapse, desc, slice

## Loading required package: GenomicRanges

## Loading required package: GenomeInfoDb

## Loading required package: SummarizedExperiment

## 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")'.

library(gplots)

##
## Attaching package: 'gplots'

## The following object is masked from 'package:IRanges':
##      space

## The following object is masked from 'package:S4Vectors':
##      space

## The following object is masked from 'package:stats':
##      lowess

library(dplyr)
library(RColorBrewer)
library(stringr)
library(genefilter)
library(data.table)

```

```

## 
## Attaching package: 'data.table'
## The following object is masked from 'package:SummarizedExperiment':
##   shift
## The following object is masked from 'package:GenomicRanges':
##   shift
## The following object is masked from 'package:IRanges':
##   shift
## The following objects are masked from 'package:S4Vectors':
##   first, second
## The following objects are masked from 'package:reshape2':
##   dcast, melt
## The following objects are masked from 'package:dplyr':
##   between, first, last
library(genefilter)
library(ggrepel)
library(viridis)

## Loading required package: viridisLite
source("http://bioconductor.org/biocLite.R")

## Bioconductor version 3.3 (BiocInstaller 1.22.3), ?biocLite for help
## A newer version of Bioconductor is available for this version of R,
##   ?BiocUpgrade for help
biocLite("org.Mm.eg.db", suppressUpdates = TRUE)

## BioC_mirror: https://bioconductor.org
## Using Bioconductor 3.3 (BiocInstaller 1.22.3), R 3.3.3 (2017-03-06).
## Installing package(s) 'org.Mm.eg.db'
## installing the source package 'org.Mm.eg.db'
require(org.Mm.eg.db)

## Loading required package: org.Mm.eg.db
## Loading required package: AnnotationDbi
##
## Attaching package: 'AnnotationDbi'
## The following object is masked from 'package:dplyr':
##   select
##

```

Pulling in the mouse counts files

```
##Pulling in the counts of only mouse genes as determined previously in my
##DGE_sorting analysis.
mousecounts <- "All mouse genes"
mouse_sampleCounts <- basename(Sys.glob(file.path(mousecounts, "*.txt")))

##Function to read in the feature counts
exptcounts <- function(files) {
  d <- read.table(files)
  d
}

##Read in all of the count files
mousecounts_readin <- lapply(file.path(mousecounts, mouse_sampleCounts), exptcounts)
names(mousecounts_readin) <- sub('humanHBVgenes.txt', '', mouse_sampleCounts)
names(mousecounts_readin)

## [1] "BD330_Ctrl_D28mouse.txt"      "BD330_Ctrl_D8mouse.txt"
## [3] "BD330_HBV_D28mouse.txt"       "BD330_HBV_D8mouse.txt"
## [5] "BD330_HBV_HDV_D28_bmouse.txt" "BD330_HBV_HDV_D28mouse.txt"
## [7] "BD330_HBV_HDV_D8_amouse.txt"  "BD330_HBV_HDV_D8mouse.txt"
## [9] "BD405A_Ctrl_D28mouse.txt"     "BD405A_Ctrl_D8mouse.txt"
## [11] "BD405A_HBV_D28mouse.txt"      "BD405A_HBV_D8mouse.txt"
## [13] "BD405A_HBV_HDV_D28mouse.txt" "BD405A_HBV_HDV_D8mouse.txt"
## [15] "Ctrl_D28_sample_1mouse.txt"   "Ctrl_D28_sample_2mouse.txt"
## [17] "Ctrl_D28_sample_3mouse.txt"   "Ctrl_D8_sample_1mouse.txt"
## [19] "Ctrl_D8_sample_2mouse.txt"    "Ctrl_D8_sample_3mouse.txt"
## [21] "HBV_D28_sample_1mouse.txt"    "HBV_D28_sample_2mouse.txt"
## [23] "HBV_D28_sample_3mouse.txt"    "HBV_D8_sample_1mouse.txt"
## [25] "HBV_D8_sample_2mouse.txt"     "HBV_D8_sample_3mouse.txt"
## [27] "HU1016_BD_co_D28mouse.txt"   "HU1016_BD_co_D8mouse.txt"
## [29] "HU1016_B_D28mouse.txt"       "HU1016_B_D8mouse.txt"
```

Now subset feature counts by “treatment” for DGE analysis.

```
ctrl <- names(mousecounts_readin)[grep("Ctrl", names(mousecounts_readin))]
ctrl_counts <- mousecounts_readin[match(ctrl, names(mousecounts_readin))]
names(ctrl_counts)

## [1] "BD330_Ctrl_D28mouse.txt"      "BD330_Ctrl_D8mouse.txt"
## [3] "BD405A_Ctrl_D28mouse.txt"     "BD405A_Ctrl_D8mouse.txt"
## [5] "Ctrl_D28_sample_1mouse.txt"   "Ctrl_D28_sample_2mouse.txt"
## [7] "Ctrl_D28_sample_3mouse.txt"   "Ctrl_D8_sample_1mouse.txt"
## [9] "Ctrl_D8_sample_2mouse.txt"    "Ctrl_D8_sample_3mouse.txt"

HBV <- names(mousecounts_readin)[grep("HBV_D|B_", names(mousecounts_readin))]
HBV_counts <- mousecounts_readin[match(HBV, names(mousecounts_readin))]
names(HBV_counts)

## [1] "BD330_HBV_D28mouse.txt"       "BD330_HBV_D8mouse.txt"
## [3] "BD405A_HBV_D28mouse.txt"      "BD405A_HBV_D8mouse.txt"
## [5] "HBV_D28_sample_1mouse.txt"    "HBV_D28_sample_2mouse.txt"
## [7] "HBV_D28_sample_3mouse.txt"    "HBV_D8_sample_1mouse.txt"
## [9] "HBV_D8_sample_2mouse.txt"     "HBV_D8_sample_3mouse.txt"
## [11] "HU1016_B_D28mouse.txt"        "HU1016_B_D8mouse.txt"
```

```

coinf <- names(mousecounts_readin)[grep("*HBV_HDV|_co_", names(mousecounts_readin))]
coinf_counts <- mousecounts_readin[match(coinf, names(mousecounts_readin))]
names(coinf_counts)

## [1] "BD330_HBV_HDV_D28_bmouse.txt" "BD330_HBV_HDV_D28mouse.txt"
## [3] "BD330_HBV_HDV_D8_amouse.txt"   "BD330_HBV_HDV_D8mouse.txt"
## [5] "BD405A_HBV_HDV_D28mouse.txt"   "BD405A_HBV_HDV_D8mouse.txt"
## [7] "HU1016_BD_co_D28mouse.txt"     "HU1016_BD_co_D8mouse.txt"

```

Make files of these separated feature counts

```

for(i in names(ctrl_counts)) {
  filename <- paste(i, sep = "")
  write.table(ctrl_counts[i], file = file.path("Mouse_d8d28_ctrl", filename),
              col.names = FALSE, row.names=FALSE,sep="\t",quote=FALSE)
}

for(i in names(HBV_counts)) {
  filename <- paste(i, sep = "")
  write.table(HBV_counts[i], file = file.path("Mouse_d8d28_HBV", filename),
              col.names = FALSE, row.names=FALSE,sep="\t",quote=FALSE)
}

for(i in names(coinf_counts)) {
  filename <- paste(i, sep = "")
  write.table(coinf_counts[i], file = file.path("Mouse_d8d28_ccoinf", filename),
              col.names = FALSE, row.names=FALSE,sep="\t",quote=FALSE)
}

```

Function to perform DGE analysis with both donor and time set as factors influencing the counts. Since we already sorted out counts into folders containing the ENSEMBL IDs for mouse genes under different infection conditions, we will pull the files from these folders to perform the DGE analysis.

```

DGE_analysis <- function(sampledirectory) {
  a <- basename(Sys.glob(file.path(sampledirectory, "*.txt")))
  sample_names <- sub('.txt', '', a)
  ##Here the donors are renamed based off the Hurel names (i.e. HU___) - RNASeq reads were all named
  ##using a different ID system.
  sampleTable <- data.frame(sampleName = sample_names, sampleFile = a, treatment =
    ifelse(grepl("Ctrl", a), "mock", ifelse(grepl("*co|*HDV", a), "coinf", "HBV")), donor =
      ifelse(grepl("BD330*", a), "HU1019", ifelse(grepl("BD405*", a), "HU1020",
                  ifelse(grepl("HU1016*", a), "HU1016", "HU1007"))), time = ifelse(grepl("*D8", a), "d8",
                  "d28"), replicate = ifelse(grepl("*sample_1|*D8_am|*D8_aa", a), "a",
                                              ifelse(grepl("*sample_2|D28_bm|D28_ba", a), "b",
                                                  ifelse(grepl("*sample_3", a), "c", ""))))
  dds <- DESeqDataSetFromHTSeqCount(sampleTable = sampleTable, directory = sampledirectory, design =
    ~donor + time)
  dds
  dds@colData
  contrast <- c("time", levels(sampleTable$time))
  output_basename <- sprintf("%s-%s_vs_%s_%s_analysis", "mousegenes", contrast[2], contrast[3], levels(sam
  dds <- estimateSizeFactors(dds)
  dds@colData
  dds <- estimateDispersions(dds)

```

```

plotDispEsts(dds, main=sprintf("%s Dispersion Estimates", output_basename))
dds <- nbinomWaldTest(dds)
res <- results(dds, contrast=contrast)
res <- res[order(res$padj, -abs(res$log2FoldChange)),]
mcols(res, use.names=TRUE)
##Log-intensity ratios = M values, log-intensity averages = A values
##Red points indicate padj < 0.1.
plotMA(res, alpha=0.1, main=sprintf(output_basename))
attr(res, "filterThreshold")

metadata(res)$alpha
metadata(res)$filterThreshold
plot(metadata(res)$filterNumRej,
  type="b", ylab="number of rejections",
  xlab="quantiles of filter")
lines(metadata(res)$lo.fit, col="red")
abline(v=metadata(res)$filterTheta)

key = "ENSEMBL"
cols = c("ENTREZID", "SYMBOL", "GENENAME", "ALIAS", "REFSEQ", "ACCCNUM")
for (col in cols) {
  # Get annotation data for column
  annotation_data <- AnnotationDbi::select(org.Mm.eg.db, rownames(res), col, keytype=key)
  # Collapse one-to-many relationships
  tmp <- aggregate(annotation_data[col], by=annotation_data[key],
    # to a list
    FUN=function(x)list(x))
  # Match on key and append to results
  idx <- match(rownames(res), tmp[[key]])
  res[[col]] <- tmp[idx,col]
}

output_data <- as.data.frame(res)
LIST_COLS <- sapply(output_data, is.list)
for (COL in colnames(output_data)[LIST_COLS]) {
  output_data[COL] <-
    sapply(output_data[COL],
      function(x)sapply(x, function(y) paste(unlist(y),
        collapse=", ") ) )
}

# Save data frame above as tab-separated file
write.table(output_data,
  file=file.path("Mouse_DGEs_donortime", paste(Sys.Date(),
                                                "mouse_donor_time",
                                                output_basename, "_results.txt", sep='')), quote=FALSE, sep="\t",
  row.names=TRUE, col.names=NA)
return(list(dds@colData, head(res)))
}

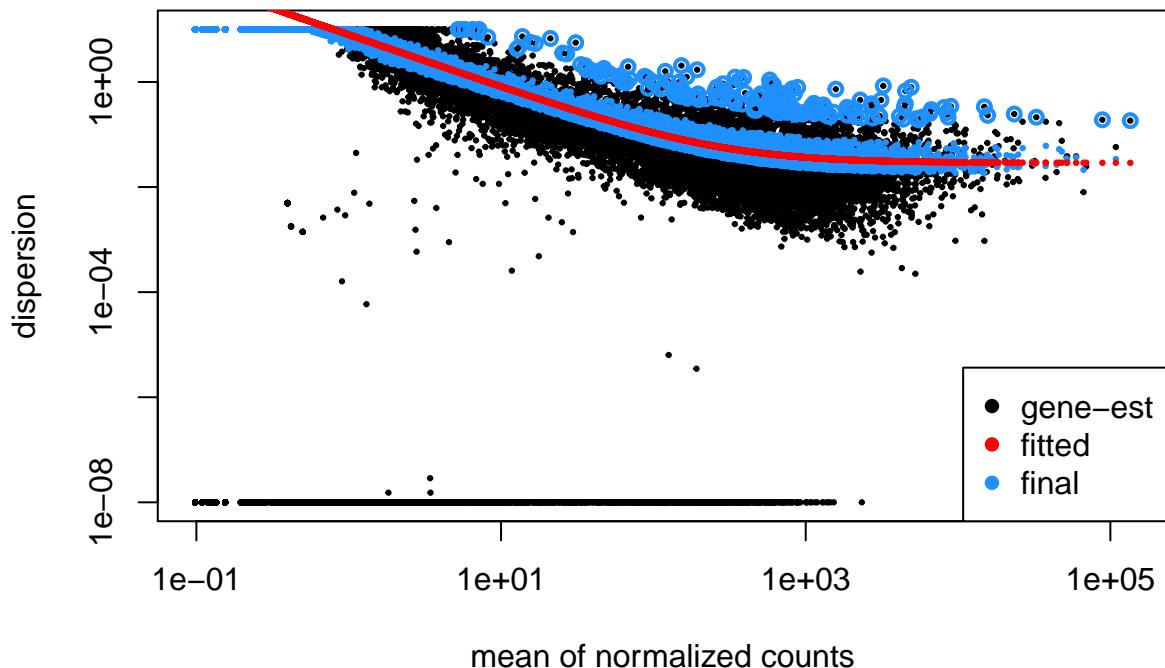
##For each infection group, determine the DGE profile when comparing
##the different times to one another (i.e. d8 versus d28).

```

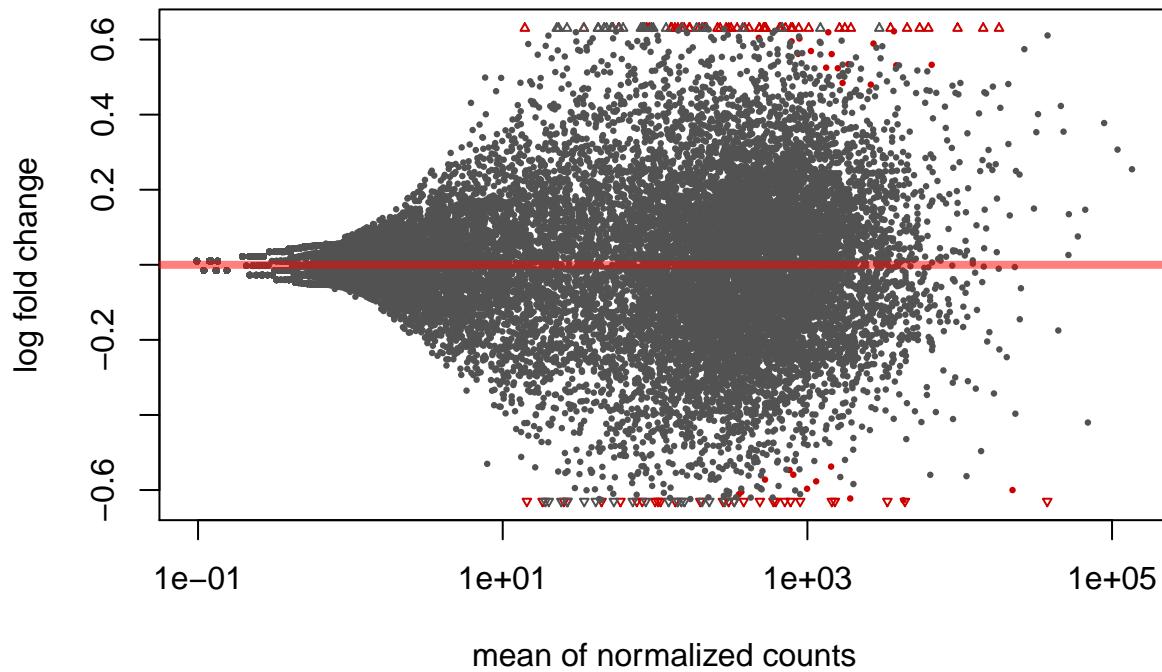
```
##coinfected  
DGE_analysis("Mouse_d8d28_coinf")
```

```
## gene-wise dispersion estimates  
## mean-dispersion relationship  
## final dispersion estimates
```

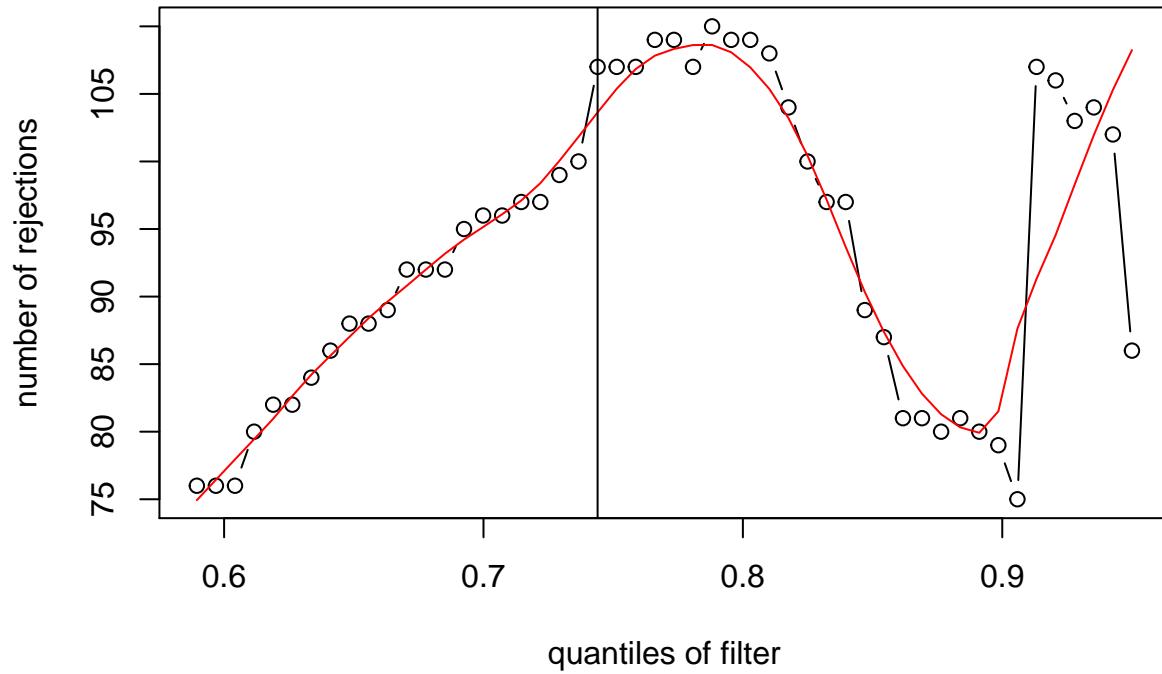
### mousegenes-d28\_vs\_d8\_coinf\_analysis Dispersion Estimates



## mousegenes-d28\_vs\_d8\_coinf\_analysis



```
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
```



```
## [[1]]
```

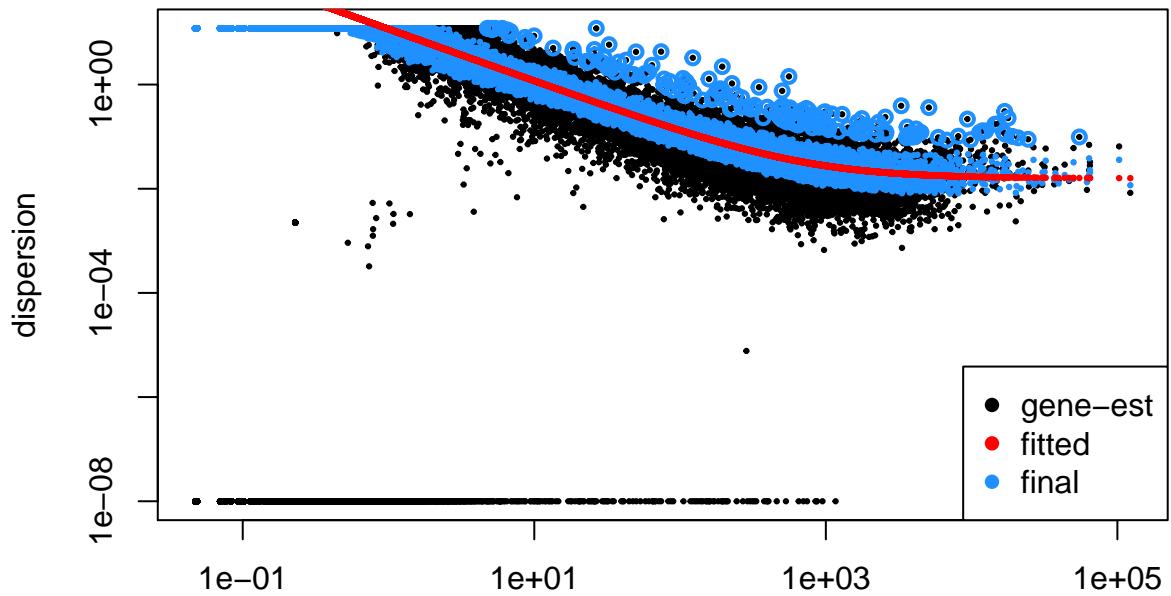
```

## DataFrame with 8 rows and 5 columns
##           treatment   donor    time replicate sizeFactor
##           <factor> <factor> <factor> <factor> <numeric>
## BD330_HBV_HDV_D28_bmouse   coinf  HU1019    d28      b  1.2732287
## BD330_HBV_HDV_D28mouse    coinf  HU1019    d28      0.9288589
## BD330_HBV_HDV_D8_amouse   coinf  HU1019     d8      a  0.8088773
## BD330_HBV_HDV_D8mouse    coinf  HU1019     d8      0.9549491
## BD405A_HBV_HDV_D28mouse   coinf  HU1020    d28      1.0221375
## BD405A_HBV_HDV_D8mouse    coinf  HU1020     d8      0.9442942
## HU1016_BD_co_D28mouse    coinf  HU1016    d28      1.0533453
## HU1016_BD_co_D8mouse     coinf  HU1016     d8      1.1453248
##
## [[2]]
## log2 fold change (MAP): time d28 vs d8
## Wald test p-value: time d28 vs d8
## DataFrame with 6 rows and 12 columns
##           baseMean log2FoldChange     lfcSE      stat
##           <numeric> <numeric> <numeric> <numeric>
## ENSMUSG00000024164 14262.79644    1.093186 0.1593001 6.862429
## ENSMUSG00000050578 126.86174     1.336499 0.2340461 5.710407
## ENSMUSG00000033453 84.92553     1.325301 0.2334195 5.677767
## ENSMUSG00000048489 91.70835     1.233146 0.2290814 5.383003
## ENSMUSG00000015647 683.07725    1.121985 0.2079982 5.394205
## ENSMUSG000000101939 4367.49714   -0.965770 0.1848971 -5.223284
##           pvalue      padj ENTREZID SYMBOL GENENAME
##           <numeric> <numeric> <list> <list> <list>
## ENSMUSG00000024164 6.769921e-12 9.266667e-08 ##### #####
## ENSMUSG00000050578 1.127064e-08 6.226425e-05 ##### #####
## ENSMUSG00000033453 1.364646e-08 6.226425e-05 ##### #####
## ENSMUSG00000048489 7.325323e-08 2.005380e-04 ##### #####
## ENSMUSG00000015647 6.882779e-08 2.005380e-04 ##### #####
## ENSMUSG000000101939 1.757772e-07 3.437197e-04 ##### #####
##           ALIAS    REFSEQ    ACCNUM
##           <list> <list> <list>
## ENSMUSG00000024164 ##### #####
## ENSMUSG00000050578 ##### #####
## ENSMUSG00000033453 ##### #####
## ENSMUSG00000048489 ##### #####
## ENSMUSG00000015647 ##### #####
## ENSMUSG000000101939 ##### #####
##monoinfected with HBV
DGE_analysis("Mouse_d8d28_HBV")

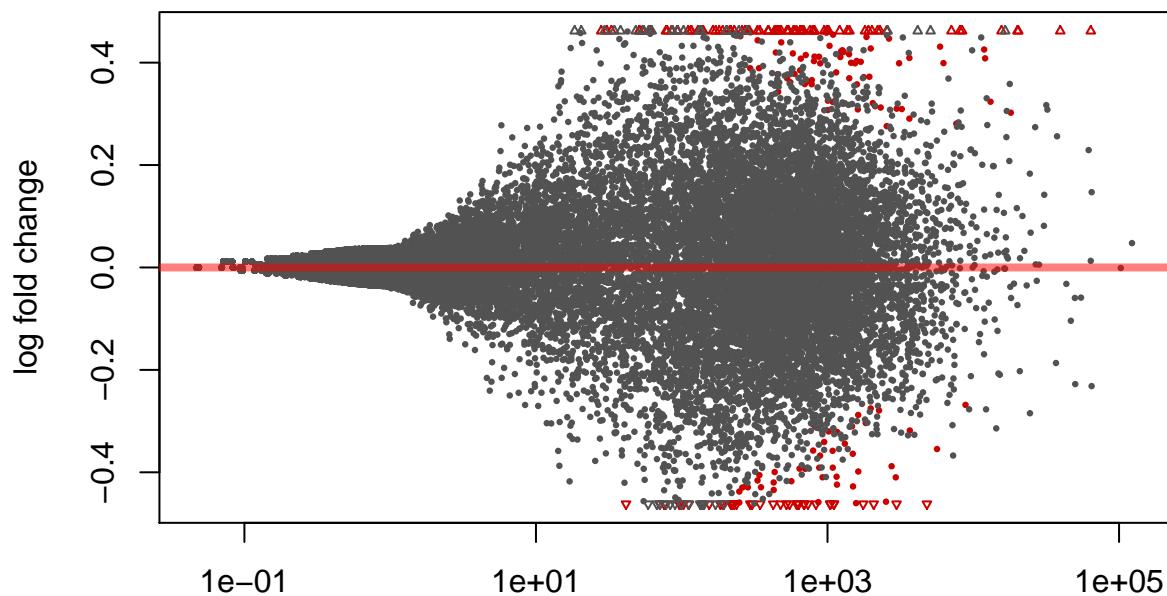
## gene-wise dispersion estimates
## mean-dispersion relationship
## final dispersion estimates

```

## mousegenes-d28\_vs\_d8\_HBV\_analysis Dispersion Estimates



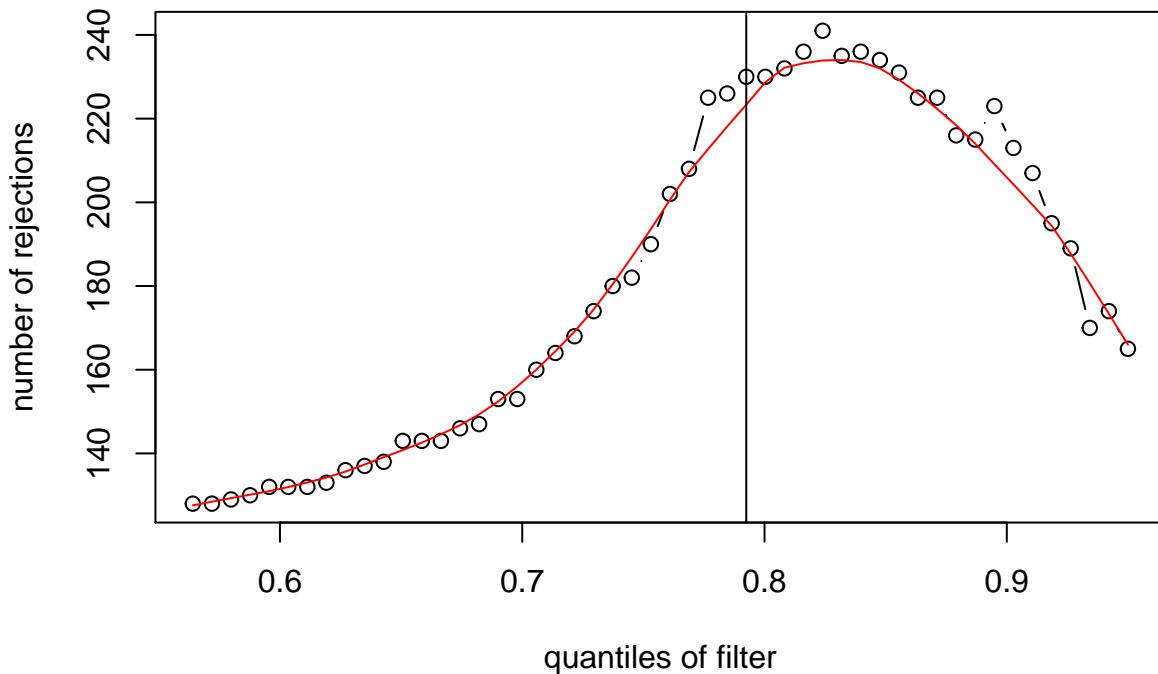
## mean of normalized counts mousegenes-d28\_vs\_d8\_HBV\_analysis



## mean of normalized counts

```
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
```

```
## 'select()' returned 1:many mapping between keys and columns
```



```
## [[1]]
```

```
## DataFrame with 12 rows and 5 columns
##   treatment donor time replicate sizeFactor
##   <factor> <factor> <factor> <factor> <numeric>
## 1 BD330_HBV_D28mouse    HBV  HU1019    d28  0.5235705
## 2 BD330_HBV_D8mouse     HBV  HU1019     d8  0.8934511
## 3 BD405A_HBV_D28mouse   HBV  HU1020    d28  0.8188988
## 4 BD405A_HBV_D8mouse   HBV  HU1020     d8  0.8670608
## 5 HBV_D28_sample_1mouse HBV  HU1007    d28      a  1.0131104
## 6 ...           ...     ...     ...      ...    ...
## 7 HBV_D8_sample_1mouse  HBV  HU1007     d8      a  1.7664648
## 8 HBV_D8_sample_2mouse  HBV  HU1007     d8      b  1.1915286
## 9 HBV_D8_sample_3mouse  HBV  HU1007     d8      c  1.7129827
## 10 HU1016_B_D28mouse   HBV  HU1016    d28  1.0908481
## 11 HU1016_B_D8mouse    HBV  HU1016     d8  0.9953479
## 12
```

```
## [[2]]
```

```
## log2 fold change (MAP): time d28 vs d8
## Wald test p-value: time d28 vs d8
## DataFrame with 6 rows and 12 columns
##   baseMean log2FoldChange      lfcSE      stat
##   <numeric>      <numeric>  <numeric>  <numeric>
## 1 ENSMUSG00000031503 15621.3787  0.5638417  7.217737
## 2 ENSMUSG00000027460  128.9846   1.1237249  7.102103
## 3 ENSMUSG00000044786  391.2283   0.8280038  6.680728
## 4 ENSMUSG00000032350  660.1810   0.8043660  6.705438
## 5 ENSMUSG00000024940  1910.3683   0.5645423  6.685466
## 6 ENSMUSG00000035673  1414.4223   0.6461405  6.485682
##   pvalue      padj ENTREZID SYMBOL GENENAME
##   <numeric>  <numeric>  <list>  <list>  <list>
## 1 5.285995e-13 5.865340e-09 ##### ##### #####
```

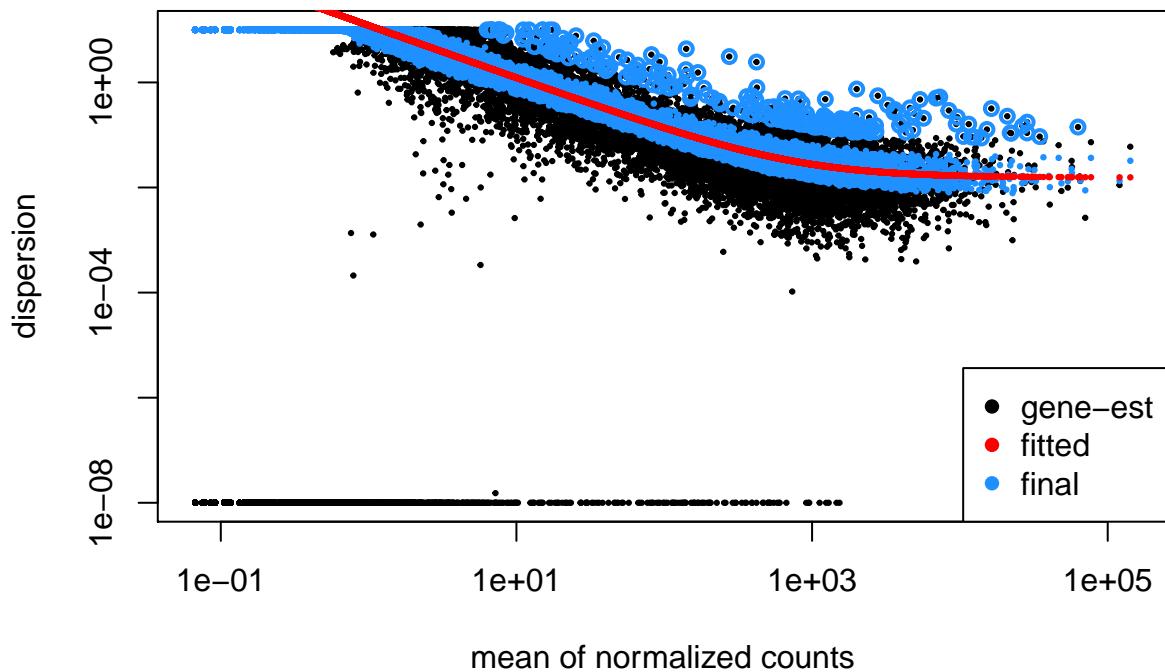
```

## ENSMUSG00000027460 1.228724e-12 6.816962e-09 ##### #####
## ENSMUSG00000044786 2.377579e-11 5.276324e-08 ##### #####
## ENSMUSG00000032350 2.008035e-11 5.276324e-08 ##### #####
## ENSMUSG00000024940 2.301906e-11 5.276324e-08 ##### #####
## ENSMUSG00000035673 8.833126e-11 1.633539e-07 ##### #####
## ALIAS      REFSEQ     ACCNUM
## <list>    <list>    <list>
## ENSMUSG00000031503 #####
## ENSMUSG00000027460 #####
## ENSMUSG00000044786 #####
## ENSMUSG00000032350 #####
## ENSMUSG00000024940 #####
## ENSMUSG00000035673 #####
##ctrl
DGE_analysis("Mouse_d8d28_ctrl")

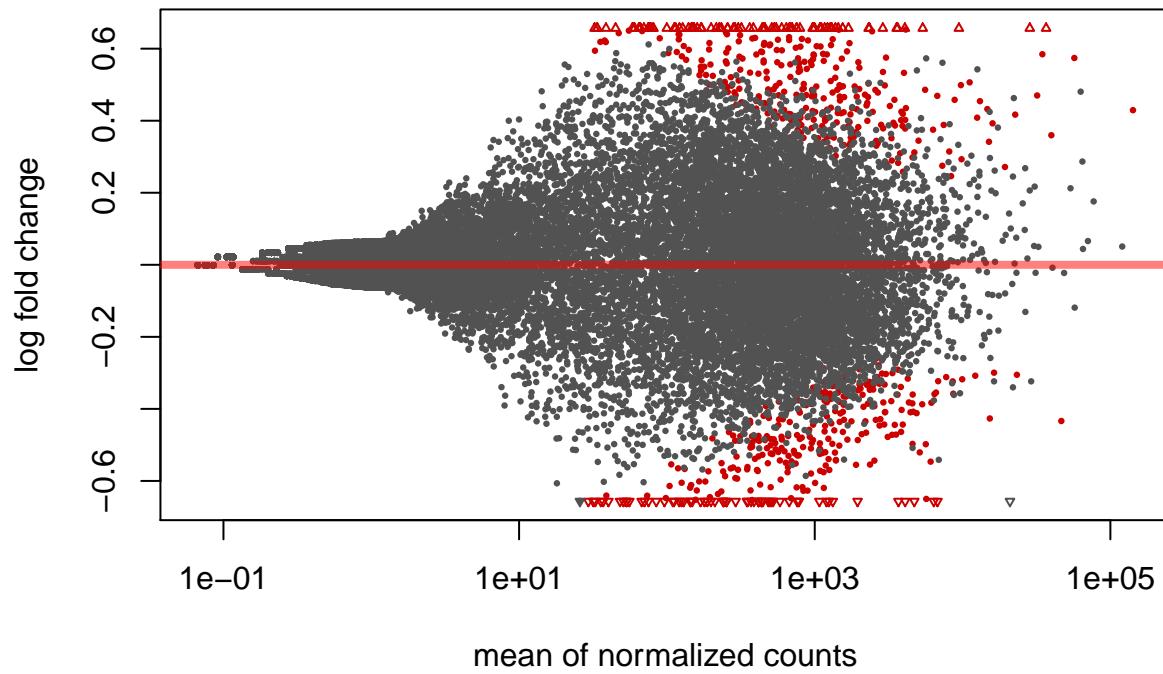
## gene-wise dispersion estimates
## mean-dispersion relationship
## final dispersion estimates

```

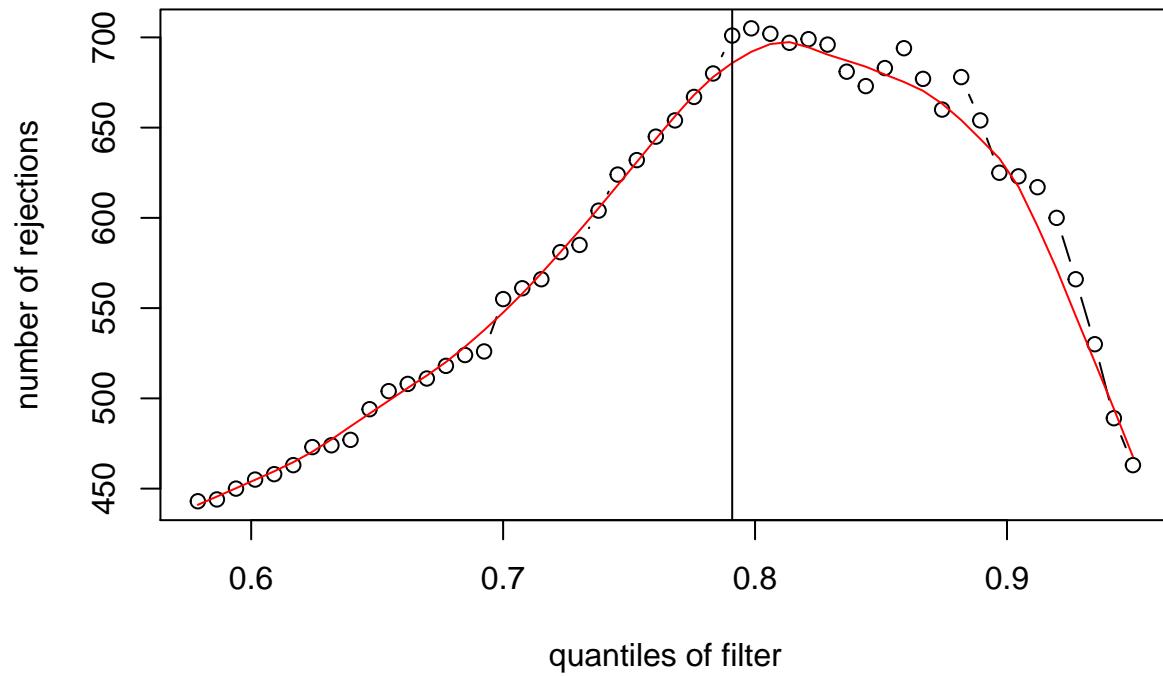
### mousegenes-d28\_vs\_d8\_mock\_analysis Dispersion Estimates



## mousegenes-d28\_vs\_d8\_mock\_analysis



```
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
```



```
## [[1]]
```

```

## DataFrame with 10 rows and 5 columns
##          treatment   donor    time replicate sizeFactor
##          <factor> <factor> <factor> <factor> <numeric>
## BD330_Ctrl_D28mouse      mock  HU1019    d28  1.0922219
## BD330_Ctrl_D8mouse       mock  HU1019     d8  1.1706558
## BD405A_Ctrl_D28mouse     mock  HU1020    d28  0.9471500
## BD405A_Ctrl_D8mouse     mock  HU1020     d8  0.8726831
## Ctrl_D28_sample_1mouse   mock  HU1007    d28     a  0.5423829
## Ctrl_D28_sample_2mouse   mock  HU1007    d28     b  0.8574041
## Ctrl_D28_sample_3mouse   mock  HU1007    d28     c  0.9023634
## Ctrl_D8_sample_1mouse   mock  HU1007     d8     a  1.3317884
## Ctrl_D8_sample_2mouse   mock  HU1007     d8     b  1.4901370
## Ctrl_D8_sample_3mouse   mock  HU1007     d8     c  1.2824368
##
## [[2]]
## log2 fold change (MAP): time d28 vs d8
## Wald test p-value: time d28 vs d8
## DataFrame with 6 rows and 12 columns
##          baseMean log2FoldChange      lfcSE      stat
##          <numeric>      <numeric> <numeric> <numeric>
## ENSMUSG00000027737  597.8454     1.2081504 0.1456778 8.293307
## ENSMUSG00000022816 6386.5856    -0.8162612 0.1003037 -8.137900
## ENSMUSG00000050578  967.1279     1.2343044 0.1528715 8.074130
## ENSMUSG00000089774 6761.5292    -1.0967639 0.1487648 -7.372470
## ENSMUSG00000032350  794.5913     0.8612594 0.1281777 6.719263
## ENSMUSG00000027460  226.2801     1.3894652 0.2085144 6.663640
##          pvalue      padj ENTREZID SYMBOL GENENAME
##          <numeric>      <numeric> <list>  <list>  <list>
## ENSMUSG00000027737 1.101427e-16 1.230955e-12 ##### #####
## ENSMUSG00000022816 4.021920e-16 2.247449e-12 ##### #####
## ENSMUSG00000050578 6.795959e-16 2.531721e-12 ##### #####
## ENSMUSG00000089774 1.674959e-13 4.679835e-10 ##### #####
## ENSMUSG00000032350 1.826458e-11 4.082499e-08 ##### #####
## ENSMUSG00000027460 2.671267e-11 4.975679e-08 ##### #####
##          ALIAS    REFSEQ    ACCNUM
##          <list>  <list>  <list>
## ENSMUSG00000027737 ##### #####
## ENSMUSG00000022816 ##### #####
## ENSMUSG00000050578 ##### #####
## ENSMUSG00000089774 ##### #####
## ENSMUSG00000032350 ##### #####
## ENSMUSG00000027460 ##### #####

```

## Session Info

```

sessionInfo()

## R version 3.3.3 (2017-03-06)
## Platform: x86_64-apple-darwin13.4.0 (64-bit)
## Running under: macOS Sierra 10.12.6
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:

```

```

## [1] parallel stats4 stats graphics grDevices utils datasets
## [8] methods base
##
## other attached packages:
## [1] org.Mm.eg.db_3.3.0 AnnotationDbi_1.34.4
## [3] BiocInstaller_1.22.3 viridis_0.4.0
## [5] viridisLite_0.2.0 ggrepel_0.6.5
## [7] data.table_1.10.0 genefilter_1.54.2
## [9] RColorBrewer_1.1-2 gplots_3.0.1
## [11] DESeq2_1.12.4 SummarizedExperiment_1.2.3
## [13] Biobase_2.32.0 GenomicRanges_1.24.3
## [15] GenomeInfoDb_1.8.7 IRanges_2.6.1
## [17] S4Vectors_0.10.3 BiocGenerics_0.18.0
## [19] openxlsx_4.0.17 reshape2_1.4.2
## [21] ggplot2_2.2.1 stringr_1.2.0
## [23] dplyr_0.7.3
##
## loaded via a namespace (and not attached):
## [1] splines_3.3.3 gtools_3.5.0 Formula_1.2-1
## [4] assertthat_0.2.0 latticeExtra_0.6-28 yaml_2.1.14
## [7] RSQLite_1.1-2 backports_1.0.5 lattice_0.20-35
## [10] glue_1.1.1 digest_0.6.12 XVector_0.12.1
## [13] checkmate_1.8.2 colorspace_1.3-2 htmltools_0.3.5
## [16] Matrix_1.2-8 plyr_1.8.4 XML_3.98-1.9
## [19] pkgconfig_2.0.1 zlibbioc_1.18.0 xtable_1.8-2
## [22] scales_0.4.1 gdata_2.17.0 BiocParallel_1.6.6
## [25] htmlTable_1.9 tibble_1.3.3 annotate_1.50.1
## [28] nnet_7.3-12 lazyeval_0.2.0 survival_2.41-3
## [31] magrittr_1.5 memoise_1.0.0 evaluate_0.10
## [34] foreign_0.8-67 tools_3.3.3 munsell_0.4.3
## [37] locfit_1.5-9.1 cluster_2.0.6 bindrcpp_0.2
## [40] caTools_1.17.1 rlang_0.1.2 grid_3.3.3
## [43] RCurl_1.95-4.8 htmlwidgets_0.9 bitops_1.0-6
## [46] base64enc_0.1-3 rmarkdown_1.4 gtable_0.2.0
## [49] DBI_0.6-1 R6_2.2.0 gridExtra_2.2.1
## [52] knitr_1.16 bindr_0.1 Hmisc_4.0-2
## [55] rprojroot_1.2 KernSmooth_2.23-15 stringi_1.1.5
## [58] Rcpp_0.12.10 geneplotter_1.50.0 rpart_4.1-10
## [61] acepack_1.4.1

```