

# Human DGEs of SACC-PHHs mono-infected with HBV or coinfected with HBV/HDV\_factors donor time\_samples up to July 2018

## Purpose:

To determine the DGE profiles (for human, HBV and HDV 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. This run includes the samples sequenced in July 2018.

```
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, which, which.max,
##   which.min

##
## 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 3.4 (BiocInstaller 1.24.0), ?biocLite for help
## A new version of Bioconductor is available after installing the most
##   recent version of R; see http://bioconductor.org/install
biocLite("org.Hs.eg.db", suppressUpdates = TRUE)

## BioC_mirror: https://bioconductor.org
## Using Bioconductor 3.4 (BiocInstaller 1.24.0), R 3.3.3 (2017-03-06).
## Installing package(s) 'org.Hs.eg.db'
## installing the source package 'org.Hs.eg.db'
require(org.Hs.eg.db)

## Loading required package: org.Hs.eg.db
## Loading required package: AnnotationDbi
##
## Attaching package: 'AnnotationDbi'

```

```

## The following object is masked from 'package:dplyr':
##
##      select
##
Pulling in the human counts files
##Pulling in the counts of only human genes as determined previously in my
##DGE_sorting analysis.
humancounts <- "All human HBV genes"
human_sampleCounts <- basename(Sys.glob(file.path(humancounts, "*.txt")))

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

##Read in all of the count files
humancounts_readin <- lapply(file.path(humancounts, human_sampleCounts), exptcounts)
names(humancounts_readin) <- sub('humanHBVgenes', '^', human_sampleCounts)
names(humancounts_readin)

## [1] "BD330 HBV_HDV Day 28 sample 1.txt"
## [2] "BD330 HBV_HDV Day 28 sample 2.txt"
## [3] "BD330 HBV_HDV Day 28 sample 3.txt"
## [4] "BD330 HBV_HDV Day 8 sample 1.txt"
## [5] "BD330 HBV_HDV Day 8 sample 2.txt"
## [6] "BD330 HBV_HDV Day 8 sample 3.txt"
## [7] "BD330_Ctrl_D28.txt"
## [8] "BD330_Ctrl_D8.txt"
## [9] "BD330_HBV_D28.txt"
## [10] "BD330_HBV_D8.txt"
## [11] "BD330_HBV_HDV_D28_b.txt"
## [12] "BD330_HBV_HDV_D28.txt"
## [13] "BD330_HBV_HDV_D8_a.txt"
## [14] "BD330_HBV_HDV_D8.txt"
## [15] "BD405A HBV_HDV D28 sample 1.txt"
## [16] "BD405A HBV_HDV D28 sample 2.txt"
## [17] "BD405A HBV_HDV D28 sample 3.txt"
## [18] "BD405A HBV_HDV D8 sample 1.txt"
## [19] "BD405A HBV_HDV D8 sample 2.txt"
## [20] "BD405A HBV_HDV D8 sample 3.txt"
## [21] "BD405A_Ctrl_D28.txt"
## [22] "BD405A_Ctrl_D8.txt"
## [23] "BD405A_HBV_D28.txt"
## [24] "BD405A_HBV_D8.txt"
## [25] "BD405A_HBV_HDV_D28.txt"
## [26] "BD405A_HBV_HDV_D8.txt"
## [27] "Ctrl_D28_sample_1.txt"
## [28] "Ctrl_D28_sample_2.txt"
## [29] "Ctrl_D28_sample_3.txt"
## [30] "Ctrl_D8_sample_1.txt"
## [31] "Ctrl_D8_sample_2.txt"
## [32] "Ctrl_D8_sample_3.txt"

```

```

## [33] "HBV_D28_sample_1.txt"
## [34] "HBV_D28_sample_2.txt"
## [35] "HBV_D28_sample_3.txt"
## [36] "HBV_D8_sample_1.txt"
## [37] "HBV_D8_sample_2.txt"
## [38] "HBV_D8_sample_3.txt"
## [39] "HU1016 Ctrl D28.txt"
## [40] "HU1016 Ctrl D8.txt"
## [41] "HU1016_BD_co_D28.txt"
## [42] "HU1016_BD_co_D8.txt"
## [43] "HU1016_B_D28.txt"
## [44] "HU1016_B_D8.txt"

```

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

```

ctrl <- names(humancounts_readin)[grep("*Ctrl", names(humancounts_readin))]
ctrl_counts <- humancounts_readin[match(ctrl, names(humancounts_readin))]
names(ctrl_counts)

## [1] "BD330_Ctrl_D28.txt"      "BD330_Ctrl_D8.txt"
## [3] "BD405A_Ctrl_D28.txt"    "BD405A_Ctrl_D8.txt"
## [5] "Ctrl_D28_sample_1.txt"   "Ctrl_D28_sample_2.txt"
## [7] "Ctrl_D28_sample_3.txt"   "Ctrl_D8_sample_1.txt"
## [9] "Ctrl_D8_sample_2.txt"    "Ctrl_D8_sample_3.txt"
## [11] "HU1016 Ctrl D28.txt"    "HU1016 Ctrl D8.txt"

HBV <- names(humancounts_readin)[grep("*HBV_D|_B_", names(humancounts_readin))]
HBV_counts <- humancounts_readin[match(HBV, names(humancounts_readin))]
names(HBV_counts)

## [1] "BD330_HBV_D28.txt"      "BD330_HBV_D8.txt"      "BD405A_HBV_D28.txt"
## [4] "BD405A_HBV_D8.txt"      "HBV_D28_sample_1.txt"  "HBV_D28_sample_2.txt"
## [7] "HBV_D28_sample_3.txt"    "HBV_D8_sample_1.txt"    "HBV_D8_sample_2.txt"
## [10] "HBV_D8_sample_3.txt"     "HU1016_B_D28.txt"     "HU1016_B_D8.txt"

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

## [1] "BD330_HBV_HDV Day 28 sample 1.txt"
## [2] "BD330_HBV_HDV Day 28 sample 2.txt"
## [3] "BD330_HBV_HDV Day 28 sample 3.txt"
## [4] "BD330_HBV_HDV Day 8 sample 1.txt"
## [5] "BD330_HBV_HDV Day 8 sample 2.txt"
## [6] "BD330_HBV_HDV Day 8 sample 3.txt"
## [7] "BD330_HBV_HDV_D28_b.txt"
## [8] "BD330_HBV_HDV_D28.txt"
## [9] "BD330_HBV_HDV_D8_a.txt"
## [10] "BD330_HBV_HDV_D8.txt"
## [11] "BD405A_HBV_HDV D28 sample 1.txt"
## [12] "BD405A_HBV_HDV D28 sample 2.txt"
## [13] "BD405A_HBV_HDV D28 sample 3.txt"
## [14] "BD405A_HBV_HDV D8 sample 1.txt"
## [15] "BD405A_HBV_HDV D8 sample 2.txt"
## [16] "BD405A_HBV_HDV D8 sample 3.txt"
## [17] "BD405A_HBV_HDV_D28.txt"
## [18] "BD405A_HBV_HDV_D8.txt"

```

```

## [19] "HU1016_BD_co_D28.txt"
## [20] "HU1016_BD_co_D8.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("HumanHBV_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("HumanHBV_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("HumanHBV_d8d28_cointf", 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 human 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),"cointf", "HBV")),
    donor = ifelse(grepl("BD330*", a), "HU1019",
      ifelse(grepl("BD405*", a), "HU1020",
        ifelse(grepl("HU1016*", a), "HU1016", "HU1007"))),
    time = ifelse(grepl("*D8|Day 8", a), "d8", "d28"),
    replicate = ifelse(grepl("*sample_1h|*D8_ah|*D8_aa|*D8_am|*sample_1m", a), "a",
      ifelse(grepl("*sample_2h|D28_bh|D28_ba|D28_bm|*sample_2m", a), "b",
        ifelse(grepl("*sample_3h| * sample 1h| * sample 1m", a), "c",
          ifelse(grepl("* sample 2h| * sample 2m", a), "d",
            ifelse(grepl("* sample 3h| * sample 3m", a), "e", ""))))))
  dds <- DESeqDataSetFromHTSeqCount(sampleTable = sampleTable, directory =
    sampledirectory, design = ~donor + time)
  dds
  dds@colData
  contrast <- c("time", levels(sampleTable$time))
  output_basename <- sprintf("%s-%s_vs_%s_%s_analysis", "humangenes", contrast[2],
    contrast[3], levels(sampleTable$treatment))
  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.Hs.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("Human_DGEs_donortime", paste(Sys.Date(),
"human_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).

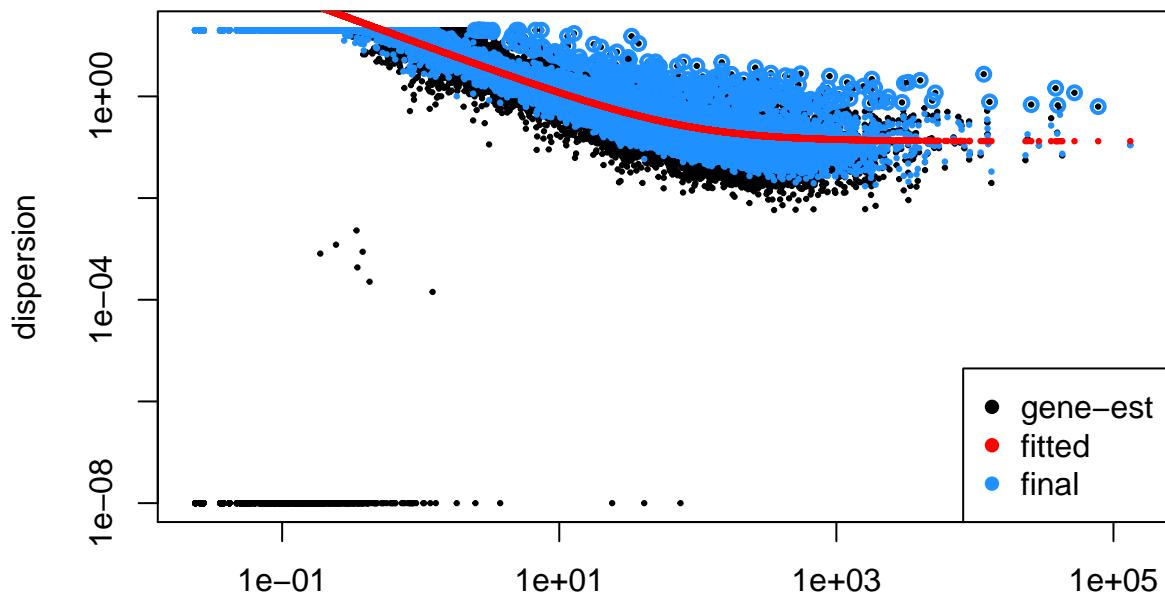
##coinfected
DGE_analysis("HumanHBV_d8d28_coinf")

## gene-wise dispersion estimates

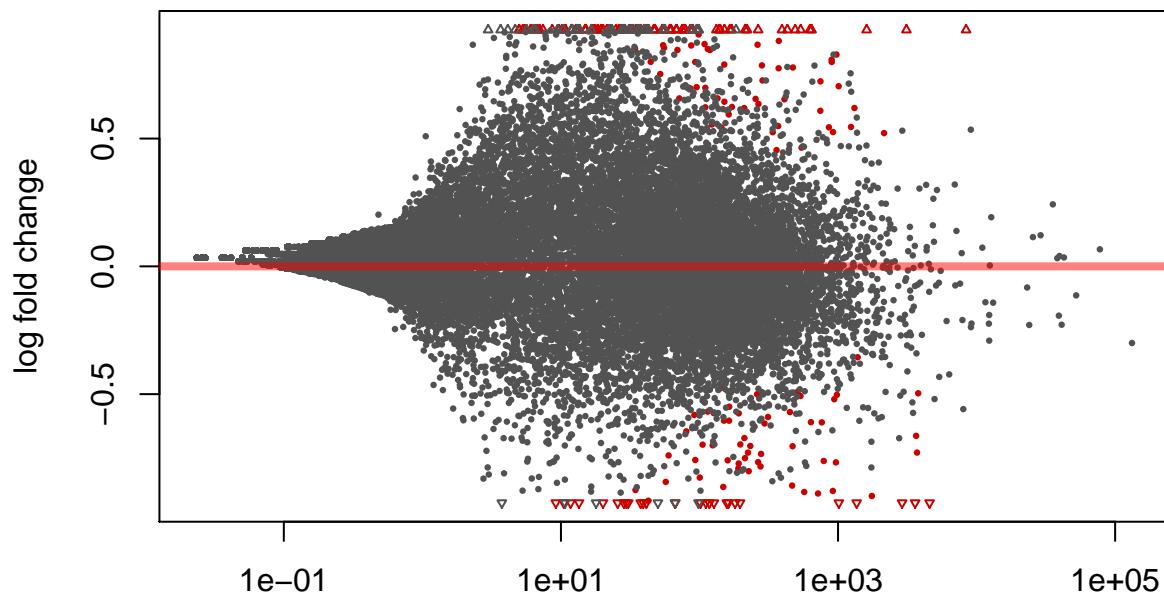
```

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

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



### mean of normalized counts humangenes-d28\_vs\_d8\_coinf\_analysis



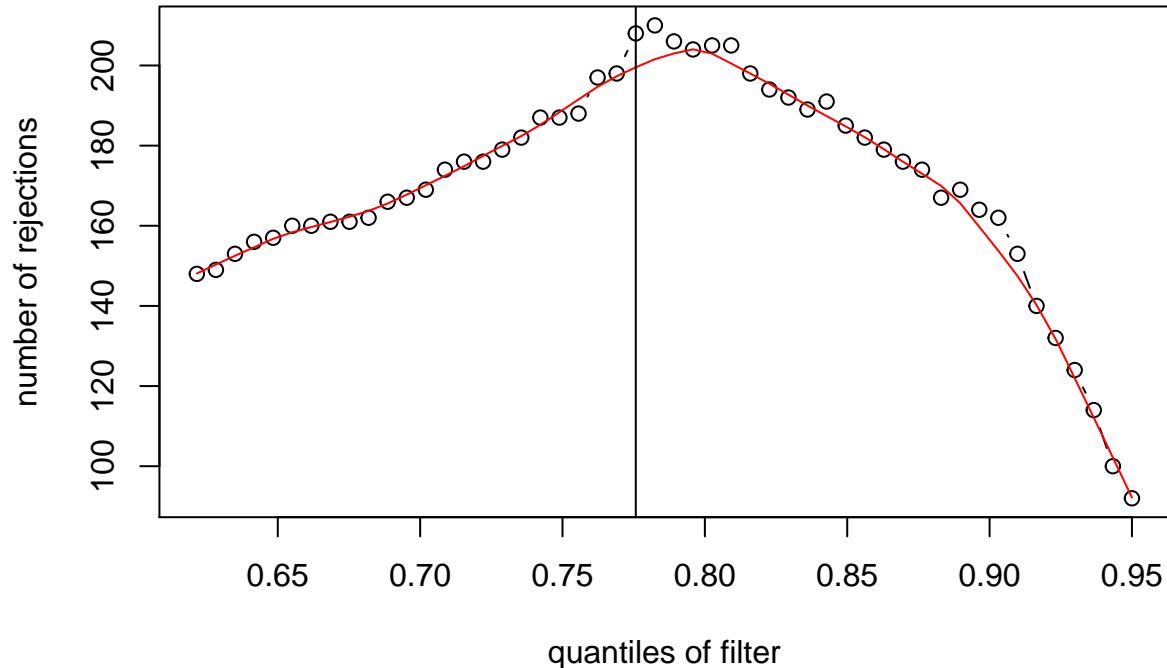
### mean of normalized counts

```
## '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 20 rows and 5 columns
##           treatment   donor    time replicate
##           <factor> <factor> <factor> <factor>
## BD330 HBV_HDV Day 28 sample 1   coinf   HU1019   d28
## BD330 HBV_HDV Day 28 sample 2   coinf   HU1019   d28
## BD330 HBV_HDV Day 28 sample 3   coinf   HU1019   d28
## BD330 HBV_HDV Day 8 sample 1    coinf   HU1019    d8
## BD330 HBV_HDV Day 8 sample 2    coinf   HU1019    d8
## ...
## ...
## BD405A HBV_HDV D8 sample 3     coinf   HU1020    d8
## BD405A_HBV_HDV_D28            coinf   HU1020   d28
## BD405A_HBV_HDV_D8            coinf   HU1020    d8
## HU1016_BD_co_D28             coinf   HU1016   d28
## HU1016_BD_co_D8              coinf   HU1016    d8
##           sizeFactor
##           <numeric>
## BD330 HBV_HDV Day 28 sample 1  0.7841005
## BD330 HBV_HDV Day 28 sample 2  0.6785288
## BD330 HBV_HDV Day 28 sample 3  0.7067862
## BD330 HBV_HDV Day 8 sample 1   1.3233612
## BD330 HBV_HDV Day 8 sample 2   1.3513499
## ...
## ...
## BD405A HBV_HDV D8 sample 3    1.4003222
## BD405A_HBV_HDV_D28          0.5666625
## BD405A_HBV_HDV_D8          0.8585337
## HU1016_BD_co_D28            0.9133729
## HU1016_BD_co_D8             0.9955121

```

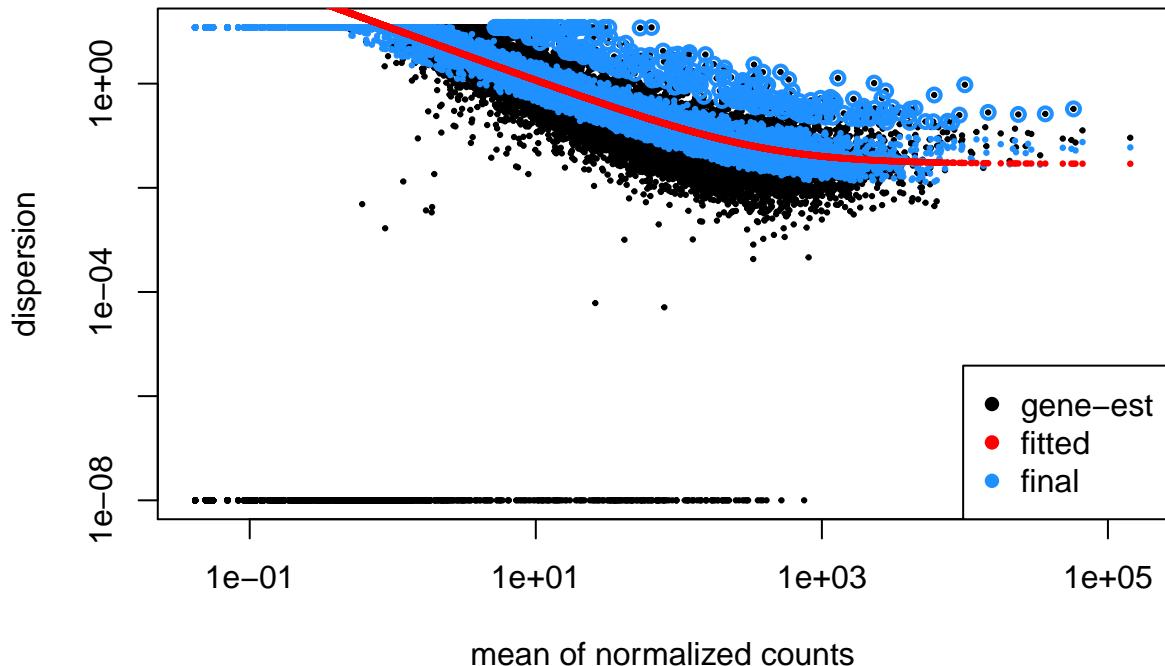
```

## 
## [[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      pvalue
##           <numeric>     <numeric> <numeric> <numeric>    <numeric>
## ENSG00000115602   85.60263      2.478735 0.2606391 9.510218 1.902634e-21
## ENSG00000145192  4579.35327     -1.486023 0.2034979 -7.302397 2.826847e-13
## ENSG00000165124  626.15900      1.223350 0.1755579 6.968355 3.206691e-12
## ENSG00000168906  394.05504      1.411852 0.2040488 6.919188 4.542402e-12
## ENSG00000164038  180.59683     -1.198910 0.1756561 -6.825324 8.772701e-12
## ENSG00000117601 1362.44656     -1.017789 0.1707905 -5.959284 2.533449e-09
##          padj ENTREZID      SYMBOL
##          <numeric>    <list> <list>
## ENSG00000115602 2.487314e-17    9173  IL1RL1
## ENSG00000145192 1.847769e-09    197   AHSG
## ENSG00000165124 1.397369e-08   79987  SVEP1
## ENSG00000168906 1.484570e-08   4144   MAT2A
## ENSG00000164038 2.293710e-08  133308  SLC9B2
## ENSG00000117601 5.519964e-06    462  SERPINC1
##                                     GENENAME
##                                     <list>
## ENSG00000115602                  interleukin 1 receptor like 1
## ENSG00000145192                  alpha 2-HS glycoprotein
## ENSG00000165124 sushi, von Willebrand factor type A, EGF and pentraxin domain containing 1
## ENSG00000168906                  methionine adenosyltransferase 2A
## ENSG00000164038                  solute carrier family 9 member B2
## ENSG00000117601                  serpin family C member 1
##          ALIAS
##          <list>
## ENSG00000115602    DER4,FIT-1,IL33R,...
## ENSG00000145192    A2HS,AHS,FETUA,...
## ENSG00000165124  C9orf13,CCP22,POLYDOM,...
## ENSG00000168906    MATA2,MATII,SAMS2,...
## ENSG00000164038    NHA2,NHE10,NHEDC2,...
## ENSG00000117601    AT3,AT3D,ATIII,...
##          REFSEQ
##          <list>
## ENSG00000115602  NM_001282408,NM_003856,NM_016232,...
## ENSG00000145192  NM_001622,NP_001613,XM_017005840,...
## ENSG00000165124                NM_153366,NP_699197
## ENSG00000168906                NM_005911,NP_005902
## ENSG00000164038  NM_001300754,NM_001300756,NM_178833,...
## ENSG00000117601  NM_000488,NP_000479,XM_005245198,...
##          ACCNUM
##          <list>
## ENSG00000115602  AAH30975,AAY15047,AB012701,...
## ENSG00000145192  AAA51683,AAB29984,AAH48198,...
## ENSG00000165124  AAH30816,AAQ89957,AAX12481,...
## ENSG00000168906  AAH01686,AAH01854,AAY24339,...
## ENSG00000164038  AA233228,AAH09732,AAH47447,...
## ENSG00000117601  AAA19930,AAB19468,AAB23132,...
```

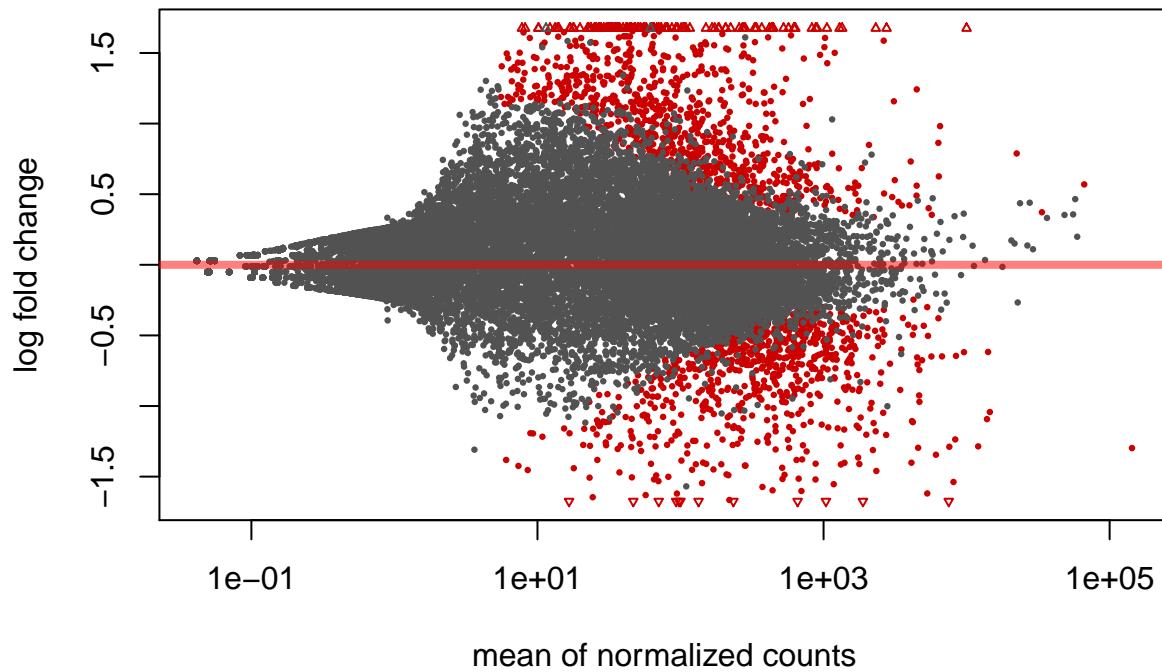
```
##monoinfected with HBV  
DGE_analysis("HumanHBV_d8d28_HBV")
```

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

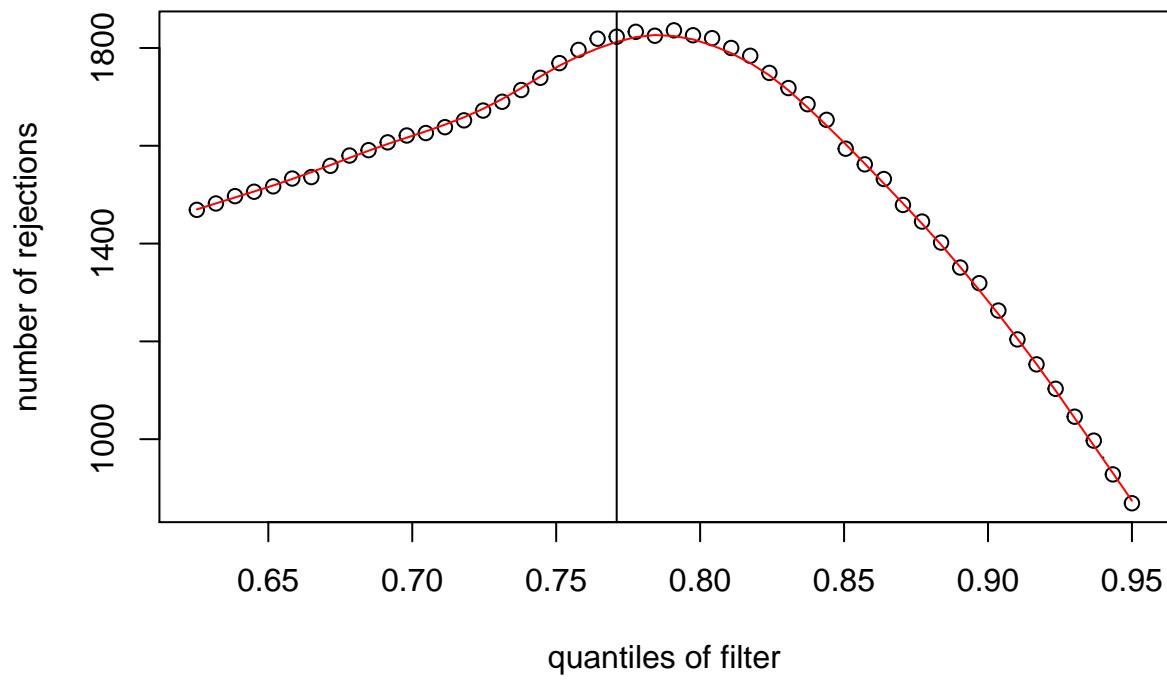
### humangenes-d28\_vs\_d8\_HBV\_analysis Dispersion Estimates



## humangenes-d28\_vs\_d8\_HBV\_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 12 rows and 5 columns
##           treatment   donor    time replicate sizeFactor
##           <factor> <factor> <factor> <factor> <numeric>
## BD330_HBV_D28       HBV   HU1019    d28      1.6059297
## BD330_HBV_D8        HBV   HU1019     d8       0.7914293
## BD405A_HBV_D28      HBV   HU1020    d28      0.3450091
## BD405A_HBV_D8      HBV   HU1020     d8       0.5533128
## HBV_D28_sample_1    HBV   HU1007    d28      1.5051295
## ...
## ...
## HBV_D8_sample_1     HBV   HU1007     d8       1.6927444
## HBV_D8_sample_2     HBV   HU1007     d8       1.1916495
## HBV_D8_sample_3     HBV   HU1007     d8       1.6385834
## HU1016_B_D28       HBV   HU1016    d28      0.5486196
## HU1016_B_D8        HBV   HU1016     d8       0.5782392
##
## [[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      pvalue
##           <numeric> <numeric> <numeric> <numeric> <numeric>
## ENSG00000087245   55.16884      3.981201  0.4028361  9.882930 4.936774e-23
## ENSG00000134871  900.68453      2.270930  0.2392537  9.491720 2.272525e-21
## ENSG00000145192 7504.92966      -1.736861 0.1981389 -8.765876 1.853299e-18
## ENSG00000171234 1178.37675      -1.553767 0.1773259 -8.762211 1.914573e-18
## ENSG00000142798 1349.82328      2.187439  0.2579493  8.480114 2.249726e-17
## ENSG00000109181  658.62636      -2.237485 0.2710106 -8.256078 1.505373e-16
##
##          padj ENTREZID SYMBOL
##          <numeric> <list> <list>
## ENSG00000087245 6.569365e-19    4313    MMP2
## ENSG00000134871 1.512024e-17    1284    COL4A2
## ENSG00000145192 6.369305e-15    197     AHSG
## ENSG00000171234 6.369305e-15    7364    UGT2B7
## ENSG00000142798 5.987422e-14    3339    HSPG2
## ENSG00000109181 3.338667e-13    7365    UGT2B10
##
##          GENENAME
##          <list>
## ENSG00000087245 matrix metallopeptidase 2
## ENSG00000134871 collagen type IV alpha 2 chain
## ENSG00000145192 alpha 2-HS glycoprotein
## ENSG00000171234 UDP glucuronosyltransferase family 2 member B7
## ENSG00000142798 heparan sulfate proteoglycan 2
## ENSG00000109181 UDP glucuronosyltransferase family 2 member B10
##
##          ALIAS
##          <list>
## ENSG00000087245 CLG4,CLG4A,MMP-2, ...
## ENSG00000134871 ICH,POREN2,COL4A2
## ENSG00000145192 A2HS,AHS,FETUA, ...
## ENSG00000171234 UDPGT 2B7,UDPGT 2B9,UDPGT2B7, ...
## ENSG00000142798 HSPG,PLC,PRCAN, ...
## ENSG00000109181 UDPGT2B10,UGT2B10
##
##          REFSEQ
##          <list>
## ENSG00000087245 NM_001127891,NM_001302508,NM_001302509, ...

```

```

## ENSG00000134871           NM_001846,NP_001837
## ENSG00000145192          NM_001622,NP_001613,XM_017005840,...
## ENSG00000171234          NM_001074,NM_001330719,NP_001065,...
## ENSG00000142798          NM_001291860,NM_005529,NP_001278789,...
## ENSG00000109181          NM_001075,NM_001144767,NM_001290091,...
##                                     ACCNUM
##                                     <list>
## ENSG00000087245 AAA35701,AAA52027,AAH02576,...
## ENSG00000134871 AAA52043,AAA53099,AAA58422,...
## ENSG00000145192 AAA51683,AAB29984,AAH48198,...
## ENSG00000171234 AAA36793,AAD14400,AAG01472,...
## ENSG00000142798 AA450342,AA507437,AAA52699,...
## ENSG00000109181 AAI13650,AK222839,AK222872,...

##ctrl
DGE_analysis("HumanHBV_d8d28_ctrl")

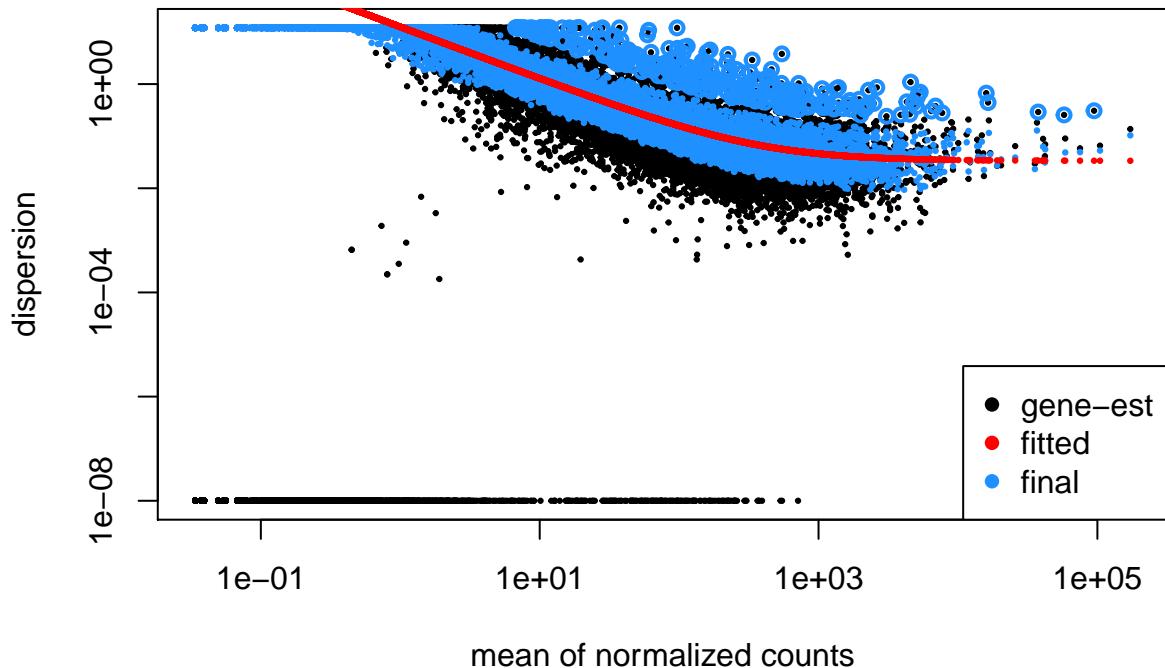
```

```

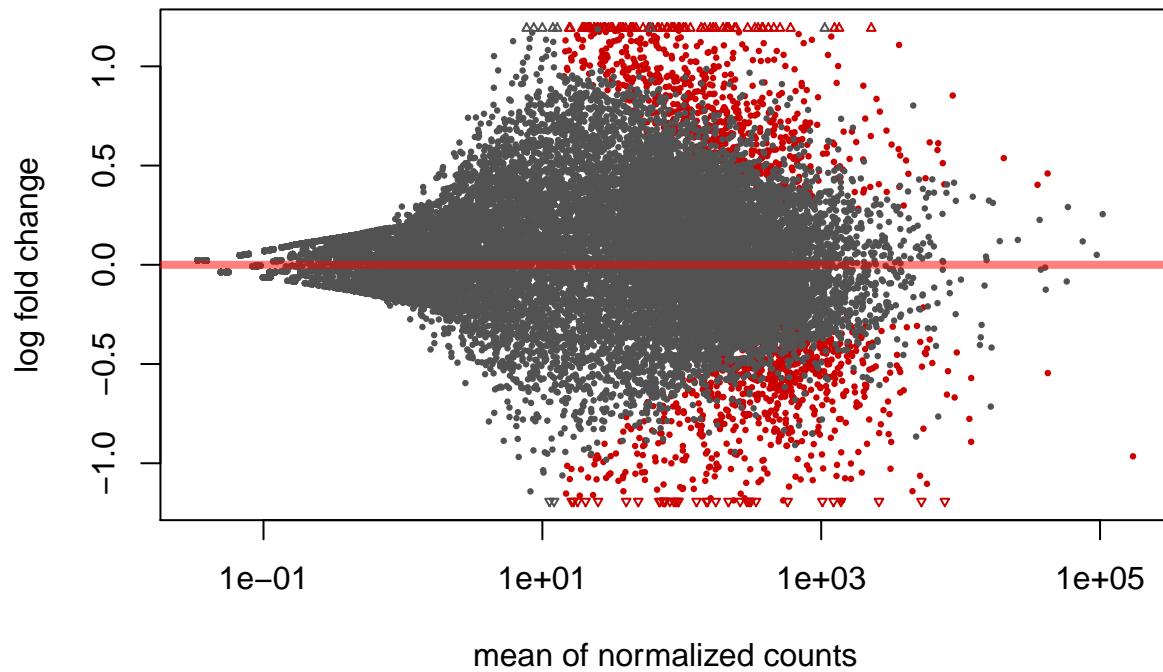
## gene-wise dispersion estimates
## mean-dispersion relationship
## final dispersion estimates

```

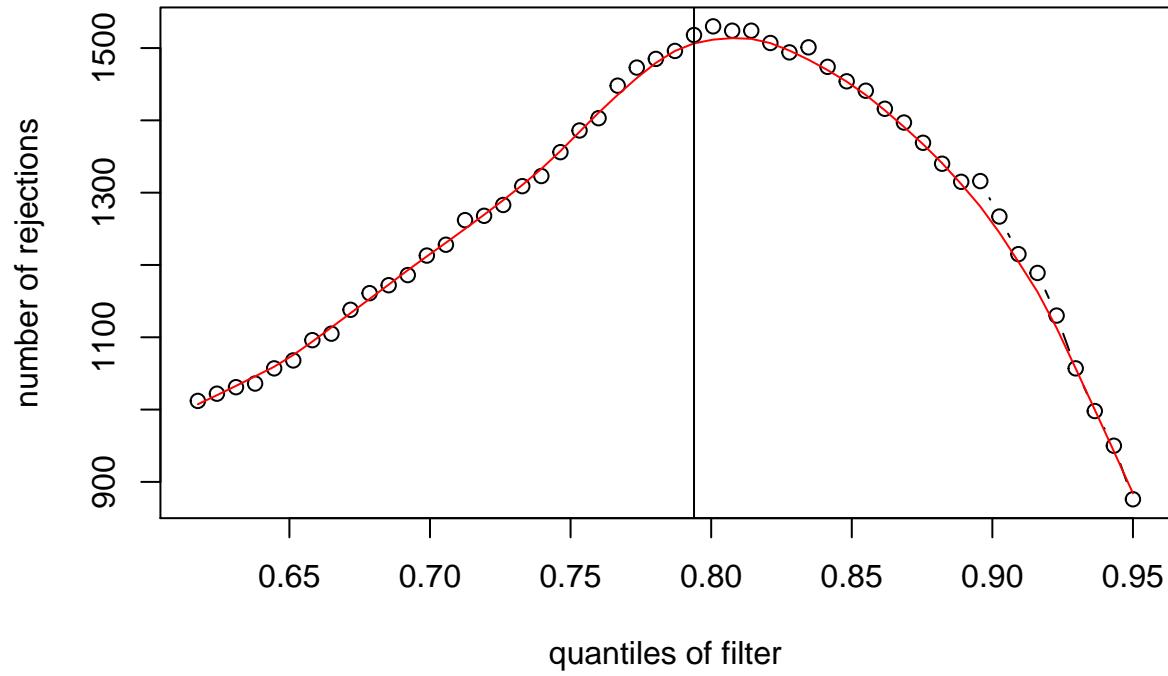
## humangenes-d28\_vs\_d8\_mock\_analysis Dispersion Estimates



## humangenes-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 12 rows and 5 columns
##           treatment   donor    time replicate sizeFactor
##           <factor> <factor> <factor> <factor> <numeric>
## BD330_Ctrl_D28      mock  HU1019    d28     0.6754286
## BD330_Ctrl_D8       mock  HU1019     d8      0.6857018
## BD405A_Ctrl_D28     mock  HU1020    d28     0.4020251
## BD405A_Ctrl_D8     mock  HU1020     d8      0.4512894
## Ctrl_D28_sample_1   mock  HU1007    d28     2.1279119
## ...                 ...    ...     ...     ...
## Ctrl_D8_sample_1   mock  HU1007     d8      1.5486329
## Ctrl_D8_sample_2   mock  HU1007     d8      1.4874719
## Ctrl_D8_sample_3   mock  HU1007     d8      1.6845046
## HU1016 Ctrl D28    mock  HU1016    d28     0.4593424
## HU1016 Ctrl D8     mock  HU1016     d8      0.7502775
##
## [[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      pvalue
##           <numeric> <numeric> <numeric> <numeric> <numeric>
## ENSG00000115602  168.1531      1.868324  0.2325148  8.035293 9.335540e-16
## ENSG00000168906  602.3687      1.397347  0.2001102  6.982888 2.891732e-12
## ENSG00000115380  376.1606      2.095007  0.3107327  6.742152 1.560572e-11
## ENSG00000249948  150.4952      -2.088258 0.3131901 -6.667700 2.598429e-11
## ENSG00000147003  313.5943      -1.240947 0.1917615 -6.471306 9.715964e-11
## ENSG00000171234 1224.3254      -1.572328 0.2509291 -6.266027 3.703772e-10
##
##          padj ENTREZID SYMBOL
##          <numeric> <list> <list>
## ENSG00000115602 1.120918e-11    9173 IL1RL1
## ENSG00000168906 1.736051e-08    4144 MAT2A
## ENSG00000115380 6.245929e-08    2202 EFEMP1
## ENSG00000249948 7.799835e-08    57733 GBA3
## ENSG00000147003 2.333192e-07    57393 TMEM27
## ENSG00000171234 7.411866e-07    7364 UGT2B7
##
##          GENENAME
##          <list>
## ENSG00000115602      interleukin 1 receptor like 1
## ENSG00000168906      methionine adenosyltransferase 2A
## ENSG00000115380      EGF containing fibulin like extracellular matrix protein 1
## ENSG00000249948      glucosylceramidase beta 3 (gene/pseudogene)
## ENSG00000147003      transmembrane protein 27
## ENSG00000171234      UDP glucuronosyltransferase family 2 member B7
##
##          ALIAS
##          <list>
## ENSG00000115602      DER4,FIT-1,IL33R, ...
## ENSG00000168906      MATA2,MATII,SAMS2, ...
## ENSG00000115380      DHRD,DRAD,FBLN3, ...
## ENSG00000249948      CBG,CBGL1,GLUC, ...
## ENSG00000147003      NX-17,NX17,TMEM27
## ENSG00000171234      UDPGT 2B7,UDPGT 2B9,UDPGT2B7, ...
##
##          REFSEQ
##          <list>
## ENSG00000115602      NM_001282408,NM_003856,NM_016232, ...

```

```

## ENSG00000168906          NM_005911,NP_005902
## ENSG00000115380 NM_001039348,NM_001039349,NM_004105,...
## ENSG00000249948 NM_001128432,NM_001277225,NM_020973,...
## ENSG00000147003      NM_020665,NP_065716,XM_017029680,...
## ENSG00000171234      NM_001074,NM_001330719,NP_001065,...
##                                     ACCNUM
##                                     <list>
## ENSG00000115602 AAH30975,AAY15047,AB012701,...
## ENSG00000168906 AAH01686,AAH01854,AAY24339,...
## ENSG00000115380 AA704122,AAA65590,AAH14410,...
## ENSG00000249948 AAG39217,AAH29362,AAH70188,...
## ENSG00000147003 AAG09466,AAH14317,AAH15099,...
## ENSG00000171234 AAA36793,AAD14400,AAG01472,...

```

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.Hs.eg.db_3.4.0      AnnotationDbi_1.36.2
## [3] BiocInstaller_1.24.0    viridis_0.5.1
## [5] viridisLite_0.3.0       ggrepel_0.7.0
## [7] data.table_1.11.0       genefilter_1.56.0
## [9] RColorBrewer_1.1-2      gplots_3.0.1
## [11] DESeq2_1.14.1           SummarizedExperiment_1.4.0
## [13] Biobase_2.34.0          GenomicRanges_1.26.4
## [15] GenomeInfoDb_1.10.3     IRanges_2.8.2
## [17] S4Vectors_0.12.2        BiocGenerics_0.20.0
## [19] openxlsx_4.0.17         reshape2_1.4.3
## [21] ggplot2_2.2.1           stringr_1.3.0
## [23] dplyr_0.7.4
##
## loaded via a namespace (and not attached):
## [1] bit64_0.9-7            splines_3.3.3      gtools_3.5.0
## [4] Formula_1.2-2          assertthat_0.2.0   latticeExtra_0.6-28
## [7] blob_1.1.1              yaml_2.1.19       pillar_1.2.2
## [10] RSQLite_2.1.0           backports_1.1.2   lattice_0.20-34
## [13] glue_1.2.0              digest_0.6.15     XVector_0.14.1
## [16] checkmate_1.8.5         colorspace_1.3-2  htmltools_0.3.6
## [19] Matrix_1.2-8            plyr_1.8.4        XML_3.98-1.11
## [22] pkgconfig_2.0.1         zlibbioc_1.20.0  xtable_1.8-2
## [25] scales_0.5.0            gdata_2.18.0     BiocParallel_1.8.2
## [28] htmlTable_1.11.2        tibble_1.4.2      annotate_1.52.1
## [31] nnet_7.3-12             lazyeval_0.2.1   survival_2.40-1

```

```
## [34] magrittr_1.5          memoise_1.1.0        evaluate_0.10.1
## [37] foreign_0.8-67         tools_3.3.3          locfit_1.5-9.1
## [40] munsell_0.4.3          cluster_2.0.5        bindrcpp_0.2.2
## [43] caTools_1.17.1         rlang_0.2.0          grid_3.3.3
## [46] RCurl_1.95-4.10        rstudioapi_0.7       htmlwidgets_1.2
## [49] bitops_1.0-6           base64enc_0.1-3      rmarkdown_1.9
## [52] gtable_0.2.0           DBI_0.8              R6_2.2.2
## [55] gridExtra_2.3          knitr_1.20          bit_1.1-12
## [58] bindr_0.1.1            Hmisc_4.1-1          rprojroot_1.3-2
## [61] KernSmooth_2.23-15     stringi_1.1.7       Rcpp_0.12.16
## [64] geneplotter_1.52.0     rpart_4.1-10         acepack_1.4.1
```