Class12_RNAseq_with_DESeq2

Zixuan Zeng(A16142927)

Table of contents


```
library(DESeq2, quietly=TRUE)
```

```
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 object is masked from 'package:utils':

    findMatches

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

    expand.grid, I, unname


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
```

```
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
```

Background

Today, we will analyze some RNAseq data from Himes et al. on the effects of a common steroid (dexmenthasone, AKA Dex) on airway smooth muscle cells (ASMs).

For this analysis we need two main inputs

- `countData`: a table of **counts** per gene (in rows) across experiments (in columns).
- `colData`: **metadata** about the design of the experiments. The rows match the columns in `countData`.

Data Import

```
counts <- read.csv("airway_scaledcounts.csv", row.names=1)
metadata <- read.csv("airway_metadata.csv", row.names=1)
```

```
head(counts)
```

```
                SRR1039508 SRR1039509 SRR1039512 SRR1039513 SRR1039516
ENSG00000000003        723        486        904        445       1170
ENSG00000000005          0          0          0          0          0
ENSG00000000419        467        523        616        371        582
```

```
ENSG00000000457        347        258        364        237        318
ENSG00000000460         96         81         73         66        118
ENSG00000000938          0          0          1          0          2
                SRR1039517 SRR1039520 SRR1039521
ENSG00000000003       1097        806        604
ENSG00000000005          0          0          0
ENSG00000000419        781        417        509
ENSG00000000457        447        330        324
ENSG00000000460         94        102         74
ENSG00000000938          0          0          0
```

```
metadata
```

```
               dex celltype     geo_id
SRR1039508 control   N61311 GSM1275862
SRR1039509 treated   N61311 GSM1275863
SRR1039512 control  N052611 GSM1275866
SRR1039513 treated  N052611 GSM1275867
SRR1039516 control  N080611 GSM1275870
SRR1039517 treated  N080611 GSM1275871
SRR1039520 control  N061011 GSM1275874
SRR1039521 treated  N061011 GSM1275875
```

Q1. How many "genes" are in this dataset?

```
nrow(counts)
```

```
[1] 38694
```

Q2. How many experiments (i.e columns in `counts` or rows in `metadata`) are in this dataset?

```
nrow(metadata)
```

```
[1] 8
```

Q3. How many "control" experiments are there in this dataset?

```
sum(metadata$dex == "control")
```

```
[1] 4
```

1. Extract the "control" columns from `counts`.
2. Calculate the mean value for each genein these columns.

3-4. Do the same for the "treated" columns.

5. compared the means between `control` and `treated` for each gene.

```
control.inds <- metadata$dex == "control"
control.counts <- counts[,control.inds]
control.means <- rowMeans(control.counts)
```

```
treated.inds <- metadata$dex == "treated"
treated.counts <- counts[,treated.inds]
treated.means <- rowMeans(treated.counts)
```

Fpr ease pf book-keeping we can store these together in one data frame called `meancounts`

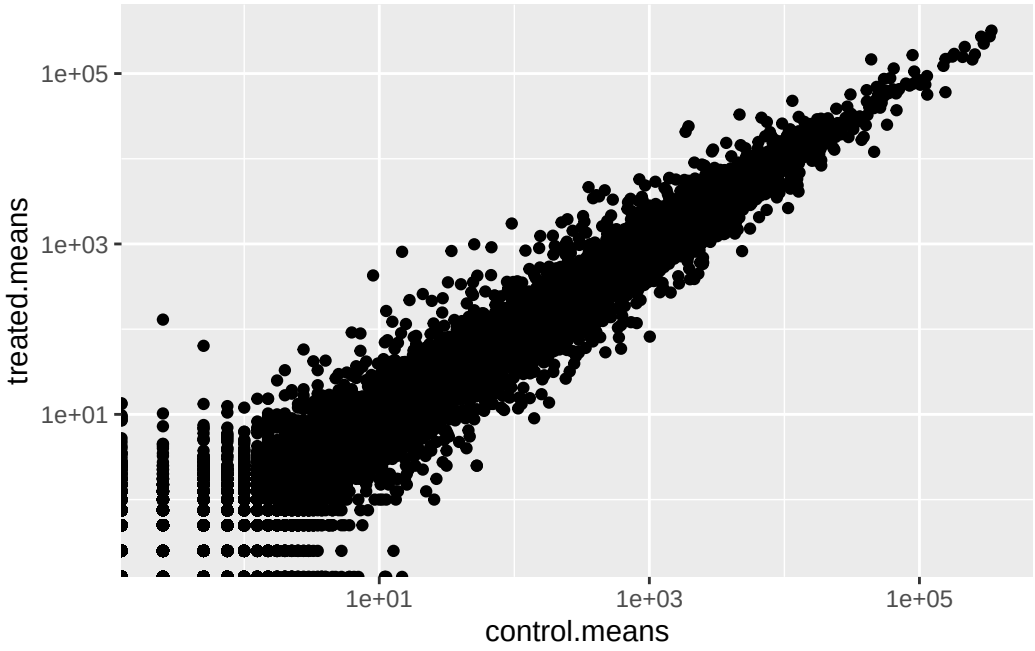
```
meancounts <- data.frame(control.means,treated.means)
head(meancounts)
```

```
                control.means treated.means
ENSG00000000003        900.75        658.00
ENSG00000000005          0.00          0.00
ENSG00000000419        520.50        546.00
ENSG00000000457        339.75        316.50
ENSG00000000460         97.25         78.75
ENSG00000000938          0.75          0.00
```

```
library(ggplot2)
ggplot(meancounts) + aes(control.means,treated.means) + geom_point() + scale_x_log10() + sca
```

```
Warning in scale_x_log10(): log-10 transformation introduced infinite values.
```

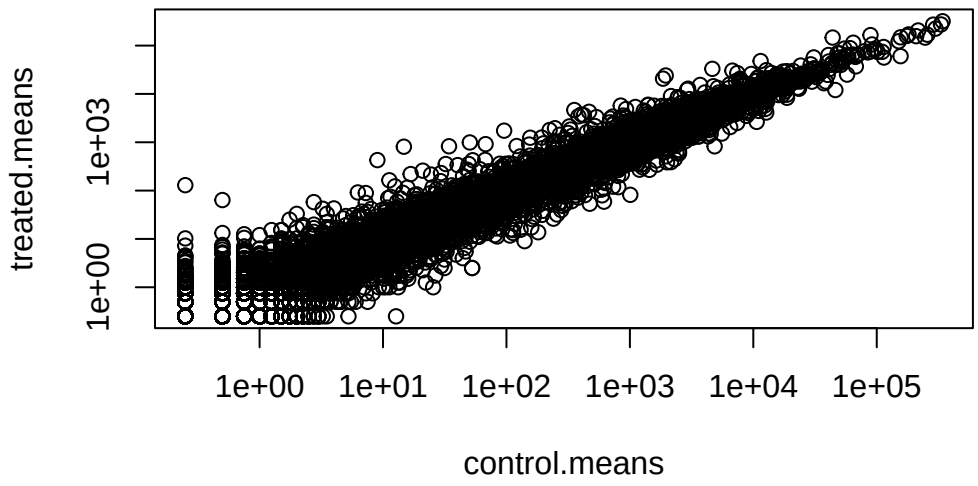
```
Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```




```
plot(meancounts,log="xy")
```

```
Warning in xy.coords(x, y, xlabel, ylabel, log): 15032 x values <= 0 omitted
from logarithmic plot
```

```
Warning in xy.coords(x, y, xlabel, ylabel, log): 15281 y values <= 0 omitted
from logarithmic plot
```



We use "fold-change" as a way to compare.

```
#treated/control
head(log2(treated.means/control.means))
```

```
ENSG00000000003 ENSG00000000005 ENSG00000000419 ENSG00000000457 ENSG00000000460
    -0.45303916             NaN      0.06900279     -0.10226805     -0.30441833
ENSG00000000938
           -Inf
```

```
meancounts$log2fc <- log2(meancounts$treated.means / meancounts$control.means)
head(meancounts)
```

```
                control.means treated.means      log2fc
ENSG00000000003        900.75        658.00 -0.45303916
ENSG00000000005          0.00          0.00         NaN
ENSG00000000419        520.50        546.00  0.06900279
ENSG00000000457        339.75        316.50 -0.10226805
ENSG00000000460         97.25         78.75 -0.30441833
ENSG00000000938          0.75          0.00        -Inf
```

A common "rule-of-thumb" threshold for calling someing "upregulated" is a log2-fold-change of +2 or more, and vice versa.

```
zero.inds <- which(meancounts[,1:2] == 0,arr.ind=TRUE)[,1]
mygenes <- meancounts[-zero.inds, ]
```

Q. How many genes are upregulated at the +2 threshold?

```
sum(mygenes$log2fc >= 2, na.rm=TRUE)
```

```
[1] 314
```

Q. How many genes are downregulated at the -2 threshold?

```
sum(mygenes$log2fc <= -2, na.rm=TRUE)
```

```
[1] 485
```

DeSeq analysis

Let's do this with DESeq2 and put some stats behind these numbers. DESeq wants 3 things for analysis: countData, colData, and design.

```
dds <- DESeqDataSetFromMatrix(countData = counts, colData = metadata, design = ~ dex)
```

```
converting counts to integer mode
```

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

The main function in the DESeq package to run analysis is called `DESeq()`.

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

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates
```

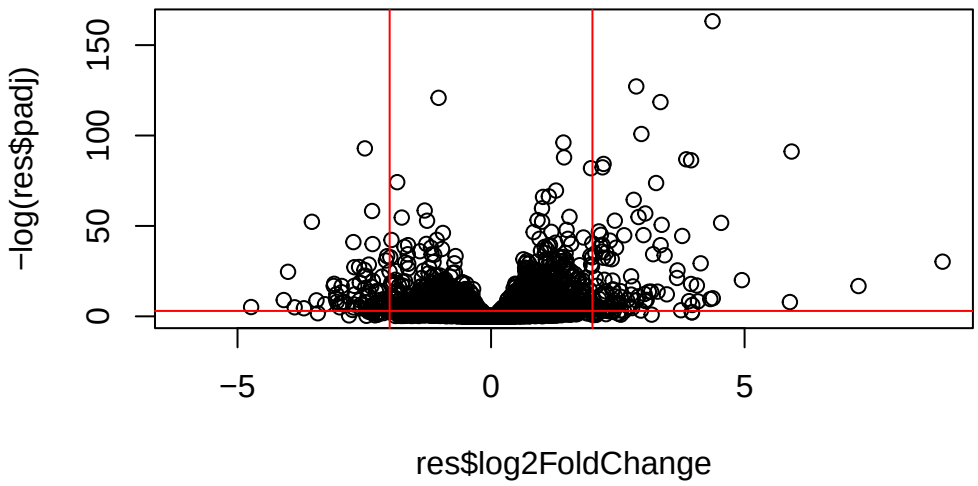
```
mean-dispersion relationship

final dispersion estimates

fitting model and testing
```

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

```
log2 fold change (MLE): dex treated vs control
Wald test p-value: dex treated vs control
DataFrame with 6 rows and 6 columns
                  baseMean log2FoldChange     lfcSE      stat    pvalue
                 <numeric>      <numeric> <numeric> <numeric> <numeric>
ENSG00000000003 747.194195      -0.350703  0.168242 -2.084514 0.0371134
ENSG00000000005   0.000000             NA        NA        NA        NA
ENSG00000000419 520.134160       0.206107  0.101042  2.039828 0.0413675
ENSG00000000457 322.664844       0.024527  0.145134  0.168996 0.8658000
ENSG00000000460  87.682625      -0.147143  0.256995 -0.572550 0.5669497
ENSG00000000938   0.319167      -1.732289  3.493601 -0.495846 0.6200029
                     padj
                <numeric>
ENSG00000000003  0.163017
ENSG00000000005        NA
ENSG00000000419  0.175937
ENSG00000000457  0.961682
ENSG00000000460  0.815805
ENSG00000000938        NA
```

Volcano Plot

```
plot(res$log2FoldChange,-log(res$padj))
abline(v=c(-2,2),col="red")
abline(h=-log(0.05),col="red")
```



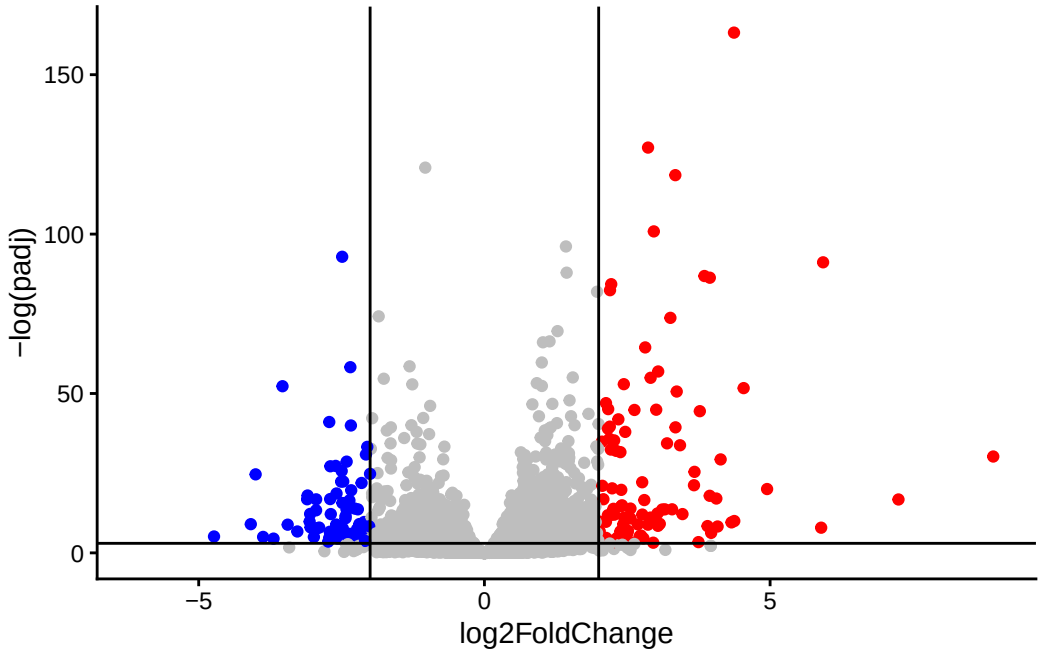
Save our results

```
write.csv(res,file="myreults.csv")
```

A nicer ggplot volcano plot

```
mycols <- rep("gray",nrow(res))
mycols[(res$log2FoldChange) > 2 ] <- "red"
mycols[(res$log2FoldChange) < -2 ] <- "blue"
mycols[res$padj >= 0.05] <- "gray"
ggplot(res) + aes(log2FoldChange,-log(padj)) + geom_point(col=mycols) +geom_vline(xintercept=
```

```
Warning: Removed 23549 rows containing missing values or values outside the scale range
(`geom_point()`).
```



Add annotation data

We need to add gene symbols, gene names and other database ids to make my results useful for further analysis.

```
library("AnnotationDbi")
library("org.Hs.eg.db")
```

Let's see what databse id formats we can translate between

```
columns(org.Hs.eg.db)
```

```
 [1] "ACCNUM"       "ALIAS"        "ENSEMBL"      "ENSEMBLPROT"  "ENSEMBLTRANS"
 [6] "ENTREZID"     "ENZYME"       "EVIDENCE"     "EVIDENCEALL"  "GENENAME"
[11] "GENETYPE"     "GO"           "GOALL"        "IPI"          "MAP"
[16] "OMIM"         "ONTOLOGY"     "ONTOLOGYALL"  "PATH"         "PFAM"
[21] "PMID"         "PROSITE"      "REFSEQ"       "SYMBOL"       "UCSCKG"
[26] "UNIPROT"
```

```
res$symbol <- mapIds(org.Hs.eg.db,
                     keys=row.names(res), # Our genenames
                     keytype="ENSEMBL",   # The format of our genenames
                     column="SYMBOL",     # The new format we want to add
                     multiVals="first")
```

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

```
head(res$symbol)
```

```
ENSG00000000003 ENSG00000000005 ENSG00000000419 ENSG00000000457 ENSG00000000460
       "TSPAN6"          "TNMD"          "DPM1"         "SCYL3"         "FIRRM"
ENSG00000000938
          "FGR"
```

Add `GENENAME` and `ENTREZID`

```
res$genename <- mapIds(org.Hs.eg.db,
                     keys=row.names(res), # Our genenames
                     keytype="ENSEMBL",   # The format of our genenames
                     column="GENENAME",   # The new format we want to add
)
```

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

```
head(res$genename)
```

```
                                                ENSG00000000003
                                                "tetraspanin 6"
                                                ENSG00000000005
                                                  "tenomodulin"
                                                ENSG00000000419
"dolichyl-phosphate mannosyltransferase subunit 1, catalytic"
                                                ENSG00000000457
                                    "SCY1 like pseudokinase 3"
                                                ENSG00000000460
  "FIGNL1 interacting regulator of recombination and mitosis"
                                                ENSG00000000938
             "FGR proto-oncogene, Src family tyrosine kinase"
```

```
res$entrezid <- mapIds(org.Hs.eg.db,
                     keys=row.names(res), # Our genenames
                     keytype="ENSEMBL",   # The format of our genenames
                     column="ENTREZID",   # The new format we want to add
)
```

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

```
head(res$entrezid)
```

```
ENSG00000000003 ENSG00000000005 ENSG00000000419 ENSG00000000457 ENSG00000000460
         "7105"          "64102"           "8813"          "57147"          "55732"
ENSG00000000938
         "2268"
```

Save my annotated results

```
write.csv(res,file = "myresults_annotated.csv")
```

Pathway analysis

We will use the **gage** function from bioconductor.

What **gage** wants as input is a named vector of importance (a vector with labeled fold-changes)

```
library(pathview)
```

```
#############################################################################
Pathview is an open source software package distributed under GNU General
Public License version 3 (GPLv3). Details of GPLv3 is available at
http://www.gnu.org/licenses/gpl-3.0.html. Particullary, users are required to
formally cite the original Pathview paper (not just mention it) in publications
or products. For details, do citation("pathview") within R.

The pathview downloads and uses KEGG data. Non-academic uses may require a KEGG
license agreement (details at http://www.kegg.jp/kegg/legal.html).
#############################################################################
```

```
library(gage)
```

```
library(gageData)

data(kegg.sets.hs)

# Examine the first 2 pathways in this kegg set for humans
head(kegg.sets.hs, 2)
```

```
$`hsa00232 Caffeine metabolism`
[1] "10"   "1544" "1548" "1549" "1553" "7498" "9"

$`hsa00983 Drug metabolism - other enzymes`
 [1] "10"     "1066"   "10720"  "10941"  "151531" "1548"   "1549"   "1551"
 [9] "1553"   "1576"   "1577"   "1806"   "1807"   "1890"   "221223" "2990"
[17] "3251"   "3614"   "3615"   "3704"   "51733"  "54490"  "54575"  "54576"
[25] "54577"  "54578"  "54579"  "54600"  "54657"  "54658"  "54659"  "54963"
[33] "574537" "64816"  "7083"   "7084"   "7172"   "7363"   "7364"   "7365"
[41] "7366"   "7367"   "7371"   "7372"   "7378"   "7498"   "79799"  "83549"
[49] "8824"   "8833"   "9"      "978"
```

```
foldchanges <- res$log2FoldChange
names(foldchanges) <- res$entrezid
head(foldchanges)
```

```
       7105       64102        8813       57147       55732        2268
-0.35070296          NA  0.20610728  0.02452701 -0.14714263 -1.73228897
```

```
data(kegg.sets.hs)
keggres = gage(foldchanges,gsets=kegg.sets.hs)
```

```
head( keggres$less, 5 )
```

```
                                                  p.geomean stat.mean
hsa05332 Graft-versus-host disease                0.0004250607 -3.473335
hsa04940 Type I diabetes mellitus                 0.0017820379 -3.002350
hsa05310 Asthma                                   0.0020046180 -3.009045
```

```
hsa04672 Intestinal immune network for IgA production 0.0060434609 -2.560546
hsa05330 Allograft rejection                          0.0073679547 -2.501416
                                                             p.val      q.val
hsa05332 Graft-versus-host disease                    0.0004250607 0.09053792
hsa04940 Type I diabetes mellitus                     0.0017820379 0.14232788
hsa05310 Asthma                                       0.0020046180 0.14232788
hsa04672 Intestinal immune network for IgA production 0.0060434609 0.31387487
hsa05330 Allograft rejection                          0.0073679547 0.31387487
                                                          set.size         exp1
hsa05332 Graft-versus-host disease                              40 0.0004250607
hsa04940 Type I diabetes mellitus                               42 0.0017820379
hsa05310 Asthma                                                 29 0.0020046180
hsa04672 Intestinal immune network for IgA production           47 0.0060434609
hsa05330 Allograft rejection                                    36 0.0073679547
```

Let's look at just one of these

```
pathview(gene.data = foldchanges,pathway.id = "hsa05310")
```

Insert figure for this pathway

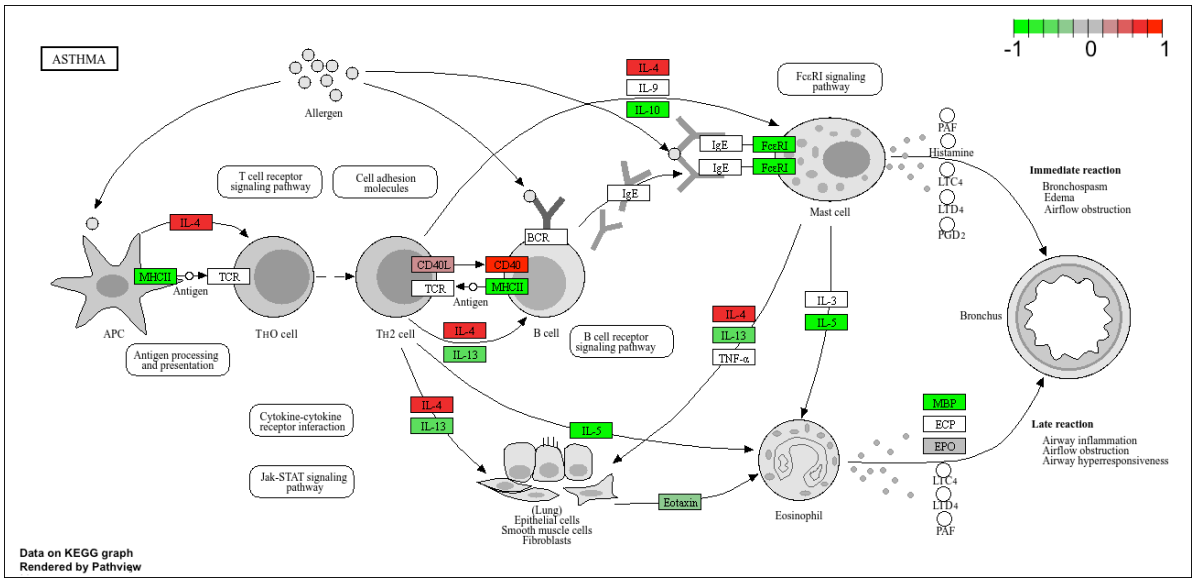


Figure 1: Asthma pathway from KEGG with my differentially expressed genes highlighted