Transcriptomics of symptomatic hosts, potato and mint, and asymptomatic host, mustard, during infection with host-adapted isolates of *Verticillim dahliae*

Authors: David Linnard Wheeler| Jeness Scott | Jeremiah Kam Sung Dung | Dennis Johnson

Table of Contents:

Objectives

Back to Table of Contents

- Characterize the differentially expressed genes involved in symptomatic (potato and mint) and asymptomatic interactions (mustard) between hosts and *Verticillium dahliae*

Hypotheses

Back to Table of Contents

- **Science H_o 1**: There are no differentially expressed genes (i) between symptomatic and asymptomatic hosts, (ii) between isolates within a host, and (iii) between hosts within an isolate.
- **Science H_o 2**: Symptomatic and asymptomatic hosts exhibit similar responses to $V.\ dahliae$ infection
- **Science H_o 3**: Gene expression of $V.\ dahliae$ does not differ accross fungal strains or between asymptomatic and symptomatic hosts
- **Statistical H_o**:
  - Observed variation in DEG across treatments represents random variation, not systematic effects of hosts or isolates Variation in the DEG is unrelated to variation in the hosts and isolates and is no greater than expected by chance or sampling error.
  - More formally:

    $$K_{ij} \sim NB(\mu_{ij}, \alpha_i)$$

    where the counts, K_{ij} for each gene, i, and sample, j, follow a negative binomial with the mean,μ_{ij}, and dispersion parameter for each gene, α_i. The dispersion parameter,α_i, describes the relationship between variance of an observed count and its mean value- the expected distance of the observed count from its mean. The mean,μ_{ij}, can be decomposed into a sample-specific size factor, s_j, and a parameter,q_{ij}, that is proportional to the expected concentation of transcripts for sample j:

    $$\mu_{ij} = s_j q_{ij}$$

    Log2 fold changes for gene i in each column of the model matrix, X,are provided by the coefficients, β_i:

    $$log_2(q_{ij}) = x_{j.}\beta_i$$
  - In short, the effect sizes between the groups are 0.

Experimental design

Back to Table of Contents

- **Treatment structure**: 2 way factorial
  - **Independent variables**:
    - 3 cultivars:
      1. Potato
      2. Mint
      3. Mustard
    - 3 fungi:
      1. $Verticillium\ dahliae\ 653$
      2. $Verticillium\ dahliae\ 111$
      3. Non-inoculated control
    - 1 time point:
      - 10 days after inoculation
    - 3 replicates
  - **Dependent variables**:

        - Constructs:
            - Gene expression
        - Variables:
            - Counts of RNA transcripts
- **Design structure**: randomized complete block design
- **Observational unit**: plant
- **Experimental unit**: plant
- **Samples**: whole plants
- **Data**:
    - RNA quantity and quality
    - Counts of RNA transcripts
- **Analysis**:
    - Differential gene expression analyses

Materials and Methods

Back to Table of Contents

- **Inoculum preparation for root dips (3.5"pot)**:
    - Inoculum for isolates 653 and 111:
        - 180 plants/3 isolates = 60 plants/isolate * 100 ml/plant (Dung et al. 2010) = 6000 ml = 6 L inoculum
        - 6 L of 10^6 conidia/ml inoculum is needed
        - 6 L/200 ml/flask = 30 flasks
    - Trial one planted: 5/1/2018
    - Trial one inoculated: 5/19/2018
    - First Harvest: potato, mint, and mustards harvested @ 10 dpi on 5/29/2018

Open data

Back to Table of Contents

- **Install and invoke packages**

In [80]:
```
if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager")
BiocManager::install("DEFormats", version = "3.8")
```

```
Bioconductor version 3.8 (BiocManager 1.30.4), R 3.5.2 (2018-12-20)
Installing package(s) 'DEFormats'


The downloaded binary packages are in
        /var/folders/8c/7fwkqlvd4ps_rj6zf9lr_xcw0000gn/T//RtmpckgoHr/down
loaded_packages

Update old packages: 'annotate', 'assertthat', 'BiocInstaller', 'callr',
'cli',
  'colorspace', 'gtable', 'highr', 'knitr', 'lazyeval', 'openssl', 'pkgbu
ild',
  'processx', 'purrr', 'Rcpp', 'RcppArmadillo', 'readxl', 'rgdal', 'rlan
g',
  'rmarkdown', 'rstudioapi', 'spam', 'sys', 'tibble', 'tinytex', 'XML',
  'diffobj', 'e1071', 'fs', 'git2r', 'Matrix', 'mgcv', 'zoo'
```

In [81]:
```
if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager")
BiocManager::install("apeglm", version = "3.8")
```

```
Bioconductor version 3.8 (BiocManager 1.30.4), R 3.5.2 (2018-12-20)
Installing package(s) 'apeglm'


The downloaded binary packages are in
        /var/folders/8c/7fwkqlvd4ps_rj6zf9lr_xcw0000gn/T//RtmpckgoHr/down
loaded_packages

Update old packages: 'annotate', 'assertthat', 'BiocInstaller', 'callr',
'cli',
  'colorspace', 'gtable', 'highr', 'knitr', 'lazyeval', 'openssl', 'pkgbu
ild',
  'processx', 'purrr', 'Rcpp', 'RcppArmadillo', 'readxl', 'rgdal', 'rlan
g',
  'rmarkdown', 'rstudioapi', 'spam', 'sys', 'tibble', 'tinytex', 'XML',
  'diffobj', 'e1071', 'fs', 'git2r', 'Matrix', 'mgcv', 'zoo'
```

In [1]:
```
library("apeglm")
library("DESeq2")
library("edgeR")
library("DEFormats")
library("dplyr")
library("ggplot2")
library("reshape2")
library("pheatmap")
library("RColorBrewer")
library("PoiClaClu")
library("ggbeeswarm")
library("EnhancedVolcano")
library("devtools")
library("gridExtra")
library("grid")
library("cowplot")
library("genefilter")
library("viridis")
library("VennDiagram")
```

```
    clusterExport, clusterMap, parApply, parCapply, parLapply,
    parLapplyLB, parRapply, parSapply, parSapplyLB

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

    IQR, mad, sd, var, xtabs

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

    anyDuplicated, append, as.data.frame, basename, cbind, colMeans,
    colnames, colSums, dirname, 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, rowMeans, rownames,
    rowSums, sapply, setdiff, sort, table, tapply, union, unique,
    unsplit, which, which.max, which.min


Attaching package: 'S4Vectors'
```

- Grab working directory

In [2]:
```
getwd()
```

'/Users/davidwheeler/Desktop/RESEARCH/Data/TRANSCRIPTOMICS/SCRIPTS'

- Set working directory

In [3]:
```
setwd("/Users/davidwheeler/Desktop/RESEARCH/Data/TRANSCRIPTOMICS/DATA/R_FIL
```

- **Open data**

In [4]:
```
DF = read.csv("Mentha_reads.csv",header=T)
```

Curate data

Back to Table of Contents

- **Set first column to index**

In [5]:
```
DF_1 <- data.frame(DF[,-1], row.names = DF[,1])
```

In [6]:
```
head(DF_1)
```

	S2_3_2_1	S2_3_2_9	S2_3_2_4	S2_1_2_1	S2_1_2_4	S2_1_2_5	S2_2_2_2	S2_2_2_5	S
Cluster-67248.142691	4.03	0.00	0.00	3.72	2.16	1.92	0.00	8.79	
Cluster-67248.107952	106.65	67.52	77.12	161.88	114.64	188.30	176.11	144.88	
Cluster-58782.0	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	
Cluster-67248.152869	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
Cluster-67248.17374	143.91	112.02	101.26	253.87	162.03	236.01	226.94	263.46	
Cluster-67248.56631	2.31	7.86	6.27	11.81	3.72	0.00	4.92	0.00	

- **Rounds floats/decimals to integer counts: since these data were generated *de novo* decimals are abound**

In [7]:
```
DF = round(DF_1, digits = 0)
```

- **Create DGEList Object**

> - Convert dataframe to matrix

In [8]:
```
df = data.matrix(DF)
```

- **Vector for column/treatment names**

In [9]:
```
group = rep(c("Control", "653", "111"), each = 3)
```

In [10]:
```
dge = DGEList(df, group = group)
```

In [11]:
```
dge
```

$counts

	S2_3_2_1	S2_3_2_9	S2_3_2_4	S2_1_2_1	S2_1_2_4	S2_1_2_5	S2_2_2_2	S2_2_2_5	S2
Cluster-67248.142691	4	0	0	4	2	2	0	9	
Cluster-67248.107952	107	68	77	162	115	188	176	145	
Cluster-58782.0	0	1	0	0	0	0	0	0	
Cluster-67248.152869	0	0	0	0	0	0	0	0	
Cluster-67248.17374	144	112	101	254	162	236	227	263	
Cluster-67248.56631	2	8	6	12	4	0	5	0	
Cluster-73865.0	0	3	0	7	6	6	0	3	

- **Coerce DGElist to DESeqDataSet**

In [12]:
```
dds = as.DESeqDataSet(dge)
```

```
converting counts to integer mode
  it appears that the last variable in the design formula, 'group',
  has a factor level, 'Control', which is not the reference level. we rec
ommend
  to use factor(...,levels=...) or relevel() to set this as the reference
level
  before proceeding. for more information, please see the 'Note on factor
levels'
  in vignette('DESeq2').
```

In [13]:
```
dds
```

```
class: DESeqDataSet
dim: 266009 9
metadata(1): version
assays(1): counts
rownames(266009): Cluster-67248.142691 Cluster-67248.107952 ...
  Cluster-67248.27096 Cluster-67248.132887
rowData names(0):
colnames(9): S2_3_2_1 S2_3_2_9 ... S2_2_2_5 S2_2_2_9
colData names(3): group lib.size norm.factors
```

Summary Statistics

Back to Table of Contents

In [ ]:

Exploratory data analyses

Back to Table of Contents

- **Transform data for pattern recognition**
- **Raw, untransformed data are used for inference downstream**

- **Filter out counts <1 to reduce dataset dimensions & expedite analysis**

In [14]:
```
dds <- dds[rowSums(counts(dds)) > 1, ]
nrow(dds)
```

246300

In [15]:
```
dds$group <- relevel(dds$group, ref = "Control")
```

- **Stabilize variance (since it is related to the mean) with variance stabilizing transformation (VST)**

In [16]:
```
vsd <- vst(dds, blind = FALSE)
```

```
-- note: fitType='parametric', but the dispersion trend was not well capt
ured by the
   function: y = a/x + b, and a local regression fit was automatically su
bstituted.
   specify fitType='local' or 'mean' to avoid this message next time.
```

- **Stabilize variance with regularized-logarithm transformation (rlog)**

In [17]:
```
rld <- rlog(dds, blind = FALSE)
```

- **Inspect the transformed data**

VST

In [18]:
```
head(assay(vsd), 3)
```

	S2_3_2_1	S2_3_2_9	S2_3_2_4	S2_1_2_1	S2_1_2_4	S2_1_2_5	S2_2_2_2	S2_2_2_5	S
Cluster-67248.142691	5.635508	5.433058	5.433058	5.561595	5.535468	5.520594	5.433058	5.689746	5
Cluster-67248.107952	7.265288	6.946230	7.119010	7.095979	7.161691	7.474105	7.370777	7.078834	7
Cluster-58782.0	5.433058	5.506769	5.433058	5.433058	5.433058	5.433058	5.433058	5.433058	5

rlog

In [19]:
```
head(assay(rld), 3)
```

	S2_3_2_1	S2_3_2_9	S2_3_2_4	S2_1_2_1	S2_1_2_4	S2_1_2_5	S2_2_2_2	S2
Cluster-67248.142691	1.771567	1.6816257	1.682194	1.7318188	1.7194922	1.7120304	1.6777742	1.8
Cluster-67248.107952	7.039000	6.7659611	6.915980	6.8921934	6.9509049	7.2132424	7.1280055	6.8
Cluster-58782.0	-0.914685	-0.8935841	-0.912875	-0.9164012	-0.9160391	-0.9162232	-0.9162492	-0.9

- **Visualize effect of transformation on data**

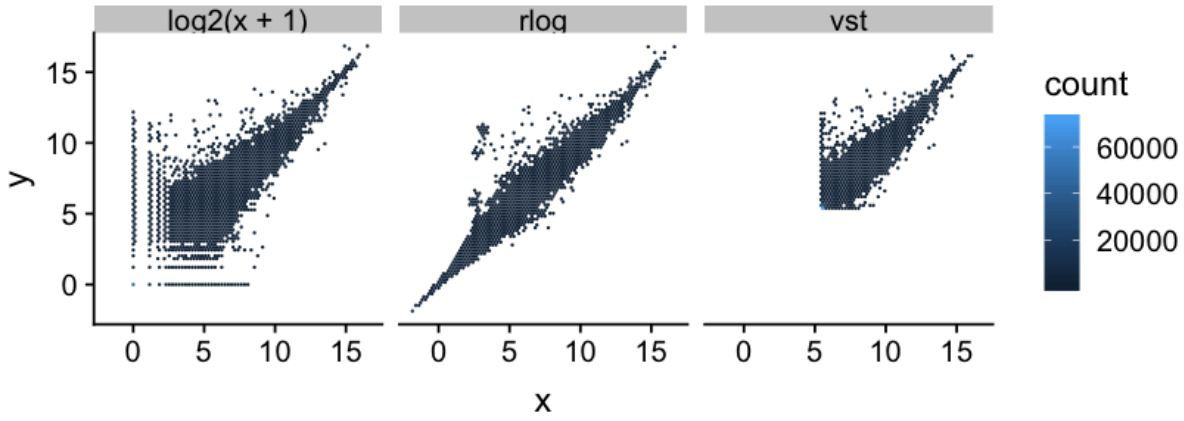
In [18]:
```
dds <- estimateSizeFactors(dds)

df <- bind_rows(
  as_data_frame(log2(counts(dds, normalized=TRUE)[, 1:2]+1)) %>%
         mutate(transformation = "log2(x + 1)"),
  as_data_frame(assay(vsd)[, 1:2]) %>% mutate(transformation = "vst"),
  as_data_frame(assay(rld)[, 1:2]) %>% mutate(transformation = "rlog"))

colnames(df)[1:2] <- c("x", "y")

ggplot(df, aes(x = x, y = y)) + geom_hex(bins = 80) +
  coord_fixed() + facet_grid( . ~ transformation)
```

```
Warning message:
“`as_data_frame()` is deprecated, use `as_tibble()` (but mind the new sem
antics).
This warning is displayed once per session.”
```



- **Compute Euclidean sample distances**

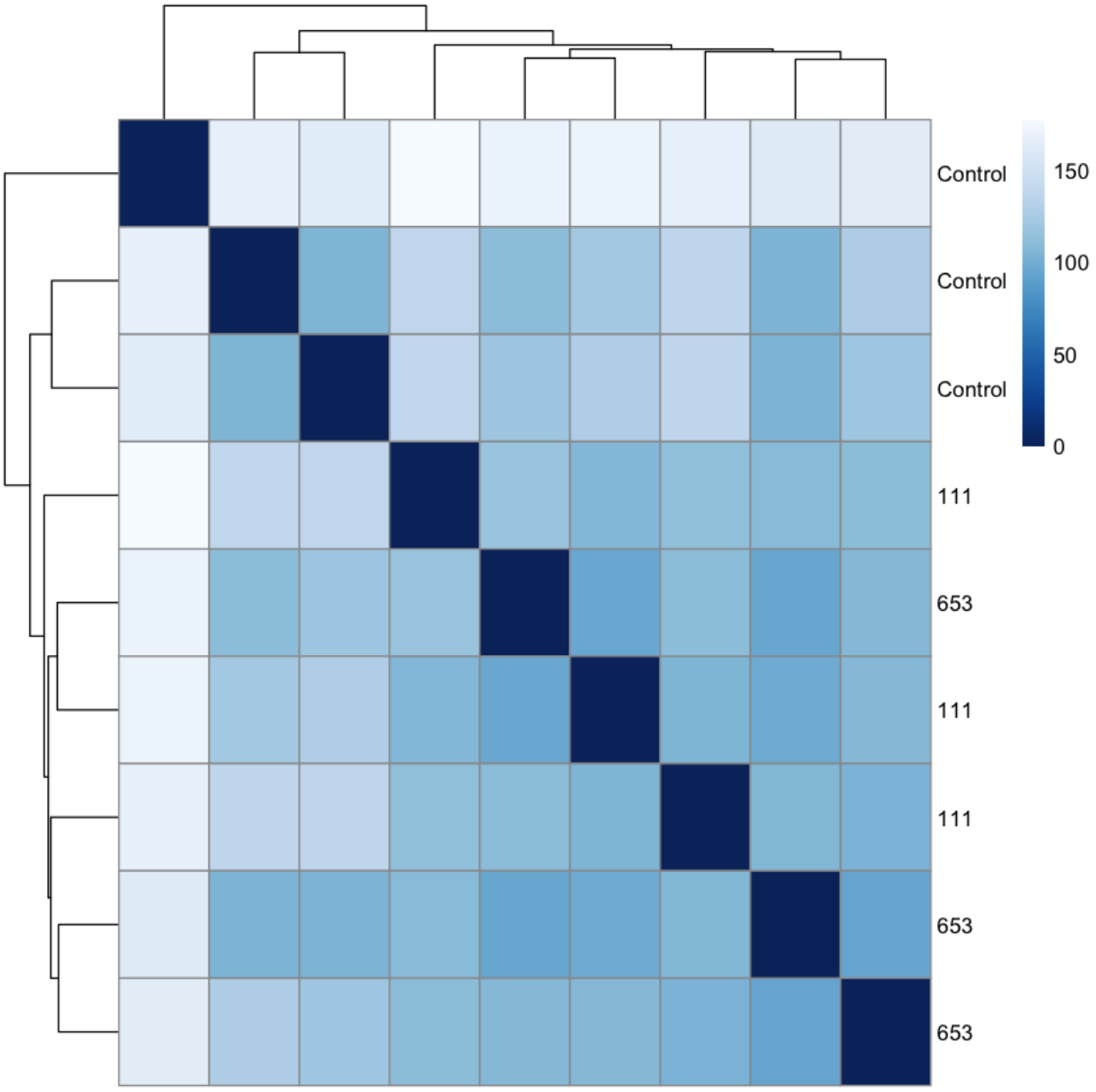
  - Transpose dataset & coerce into matrix

In [19]:
```
sampleDists <- dist(t(assay(vsd)))
sampleDists
```

```
          S2_3_2_1  S2_3_2_9  S2_3_2_4  S2_1_2_1  S2_1_2_4  S2_1_2_5  S2_
2_2_2
S2_3_2_9 167.28496
S2_3_2_4 104.57038 162.75034
S2_1_2_1 104.13654 160.54175 104.33672
S2_1_2_4 126.75408 163.34956 120.17469  93.91661
S2_1_2_5 110.18749 169.81285 118.84469  94.13099 107.49950
S2_2_2_2 136.08553 167.08227 136.59572 105.99619 103.36629 109.92072
S2_2_2_5 121.31476 171.43671 128.65036  97.71853 107.58663  95.84291 105.
08904
S2_2_2_9 138.30702 177.54195 138.01094 108.78442 110.23610 116.27587 112.
14574
          S2_2_2_5
S2_3_2_9
S2_3_2_4
S2_1_2_1
S2_1_2_4
S2_1_2_5
S2_2_2_2
S2_2_2_5
S2_2_2_9 107.05215
```

- **Visualize sample distances with heatmap using Euclidean distances**

In [20]:
```
sampleDistMatrix <- as.matrix( sampleDists )
rownames(sampleDistMatrix) <- paste( vsd$group)
colnames(sampleDistMatrix) <- NULL
colors <- colorRampPalette(rev(brewer.pal(9, "Blues")))(255)
pheatmap(sampleDistMatrix,
         clustering_distance_rows = sampleDists,
         clustering_distance_cols = sampleDists,
         cluster_rows=T, cluster_cols=T,
         col = colors)
```



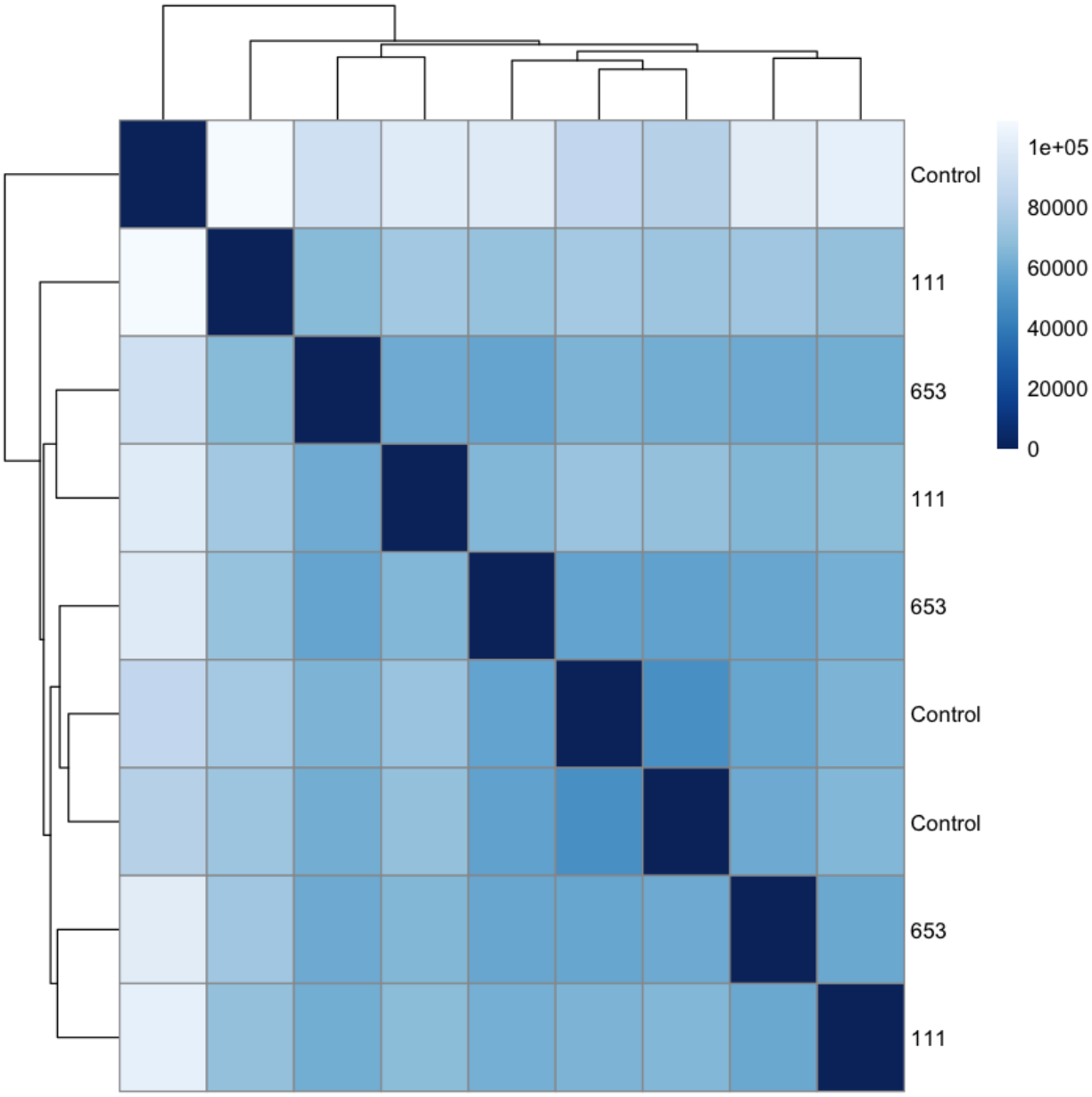
- **Compute Poisson distances (Witten 2011)**

  - Distances

In [21]:
```
poisd <- PoissonDistance(t(counts(dds)))
```

- **Visualize sample distances with heatmap using Poisson distances**

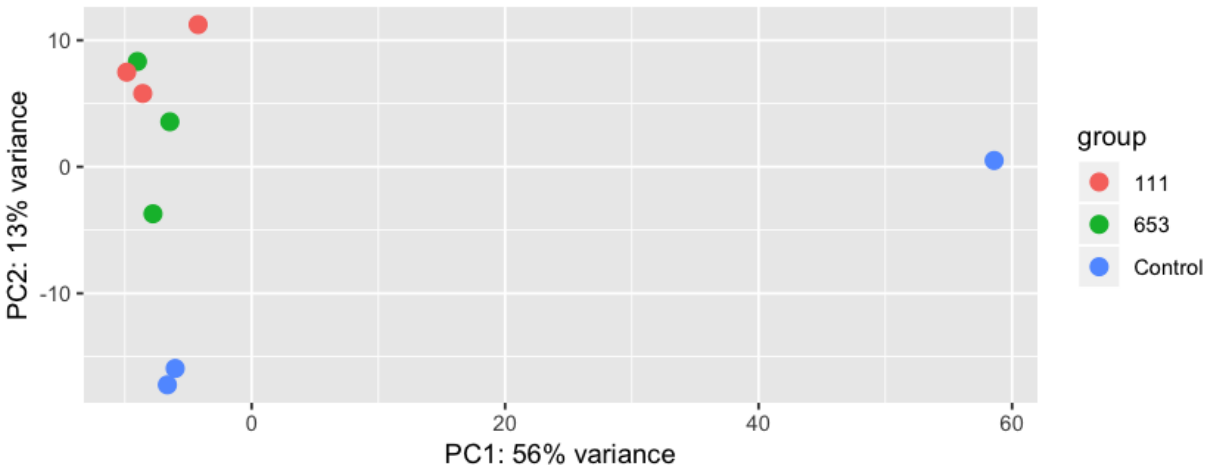
In [22]:
```
samplePoisDistMatrix <- as.matrix( poisd$dd )
rownames(samplePoisDistMatrix) <- paste( dds$group, sep=" - " )
colnames(samplePoisDistMatrix) <- NULL
pheatmap(samplePoisDistMatrix,
         clustering_distance_rows = poisd$dd,
         clustering_distance_cols = poisd$dd,
         col = colors)
```



- **Visualize sample-to-sample differences with pricipal components analysis (PCA)**

- PCA with DESeq2

In [23]:
```
plotPCA(vsd, intgroup = c("group"))
```



- PCA with ggplot2

- View PCs

In [24]:
```
pcaData <- plotPCA(vsd, intgroup = c( "group"), returnData = TRUE)
pcaData
```

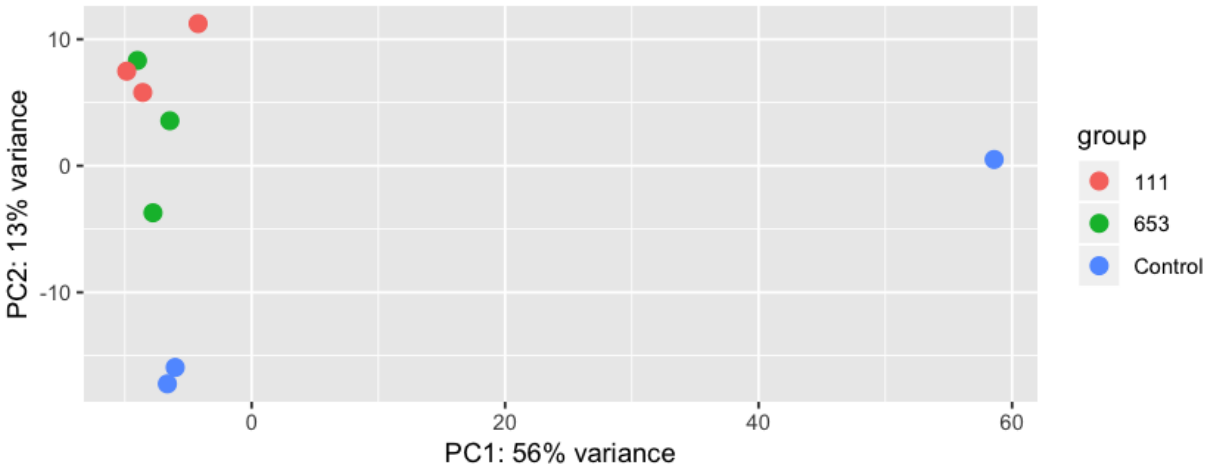
	PC1	PC2	group	group.1	name
S2_3_2_1	-6.646067	-17.2333967	Control	Control	S2_3_2_1
S2_3_2_9	58.586935	0.4998174	Control	Control	S2_3_2_9
S2_3_2_4	-6.029665	-15.9366397	Control	Control	S2_3_2_4
S2_1_2_1	-7.783224	-3.7150135	653	653	S2_1_2_1
S2_1_2_4	-6.452647	3.5560928	653	653	S2_1_2_4
S2_1_2_5	-9.011399	8.3241647	653	653	S2_1_2_5
S2_2_2_2	-4.222302	11.2345655	111	111	S2_2_2_2
S2_2_2_5	-8.588481	5.7921773	111	111	S2_2_2_5
S2_2_2_9	-9.853150	7.4782322	111	111	S2_2_2_9

- Express variation explained to percentage

In [25]:
```
percentVar <- round(100 * attr(pcaData, "percentVar"))
```

- Plot data

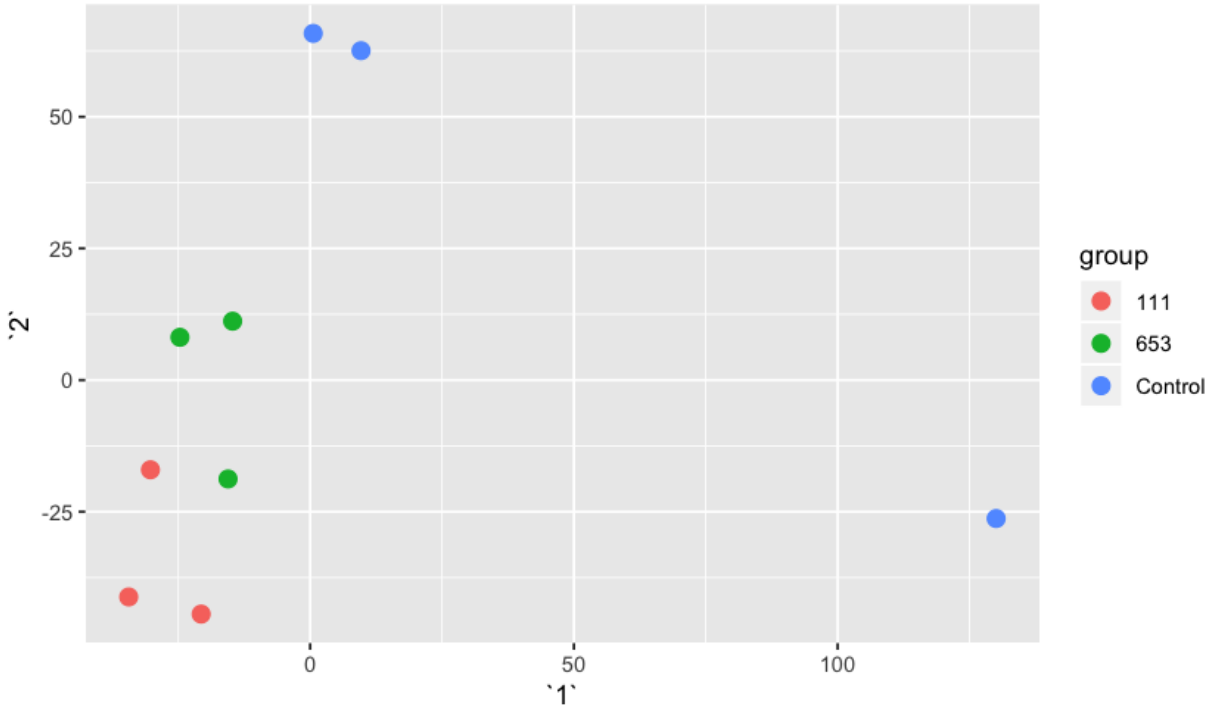
In [26]:
```
ggplot(pcaData, aes(x = PC1, y = PC2, color = group)) +
  geom_point(size =3) +
  xlab(paste0("PC1: ", percentVar[1], "% variance")) +
  ylab(paste0("PC2: ", percentVar[2], "% variance")) +
  coord_fixed()
```



- **Visualize sample-to-sample differences with multidimensional scaling (MDS)**

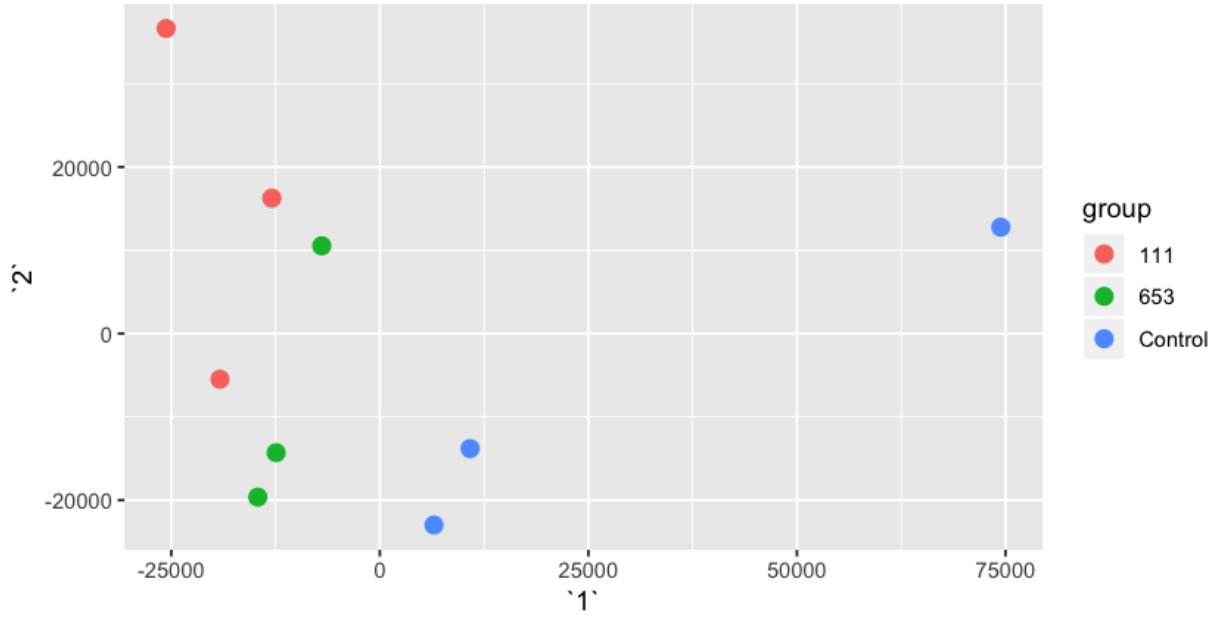
  - VSD data

In [27]:
```
mds <- as.data.frame(colData(vsd))  %>%
         cbind(cmdscale(sampleDistMatrix))
ggplot(mds, aes(x = `1`, y = `2`, color = group)) +
  geom_point(size = 3) + coord_fixed()
```



- Poisson data

In [28]:
```
mdsPois <- as.data.frame(colData(dds)) %>%
   cbind(cmdscale(samplePoisDistMatrix))
ggplot(mdsPois, aes(x = `1`, y = `2`, color = group)) +
  geom_point(size = 3) + coord_fixed()
```



Diagnostics

Back to Table of Contents

In [ ]:
```
```

Parametric analysis: differential expression analysis

Back to Table of Contents

- **Identify differentially expressed genes with raw count data**

In [19]:
```
dds <- DESeq(dds)
```

```
using pre-existing size factors
estimating dispersions
gene-wise dispersion estimates
mean-dispersion relationship
final dispersion estimates
fitting model and testing
```

- **Tabulate results, set $\alpha = 0.05$**
- **Adjust p-values with Benjamini & Hochberg (1995) to account for false discoveries**
- **Shrink/deflate effect sizes (Log fold change estimates)**

- All data

In [20]:
```
All = results(dds, independentFiltering=TRUE, alpha=0.001, pAdjustMethod="B
```

- Summary

In [21]:
```
summary(All)
```

```
out of 246300 with nonzero total read count
adjusted p-value < 0.001
LFC > 0 (up)       : 73, 0.03%
LFC < 0 (down)     : 111, 0.045%
outliers [1]       : 2543, 1%
low counts [2]     : 138421, 56%
(mean count < 6)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

- Contrast control vs 653

In [22]:
```
Cv653 = results(dds, contrast=c("group","653","Control"),
                independentFiltering=TRUE, alpha=0.001, pAdjustMethod="BH",
Cv653 = lfcShrink(dds, contrast=c("group","653","Control"), res=Cv653)
```

```
using 'normal' for LFC shrinkage, the Normal prior from Love et al (201
4).
additional priors are available via the 'type' argument, see ?lfcShrink f
or details
```

- Summary

In [23]:
```
summary(Cv653)
```

```
out of 246300 with nonzero total read count
adjusted p-value < 0.001
LFC > 0 (up)       : 73, 0.03%
LFC < 0 (down)     : 111, 0.045%
outliers [1]       : 2543, 1%
low counts [2]     : 138421, 56%
(mean count < 6)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

- Contrast control vs 111

In [24]:
```
Cv111 = results(dds, contrast=c("group","111","Control"),
                independentFiltering=TRUE, alpha=0.001, pAdjustMethod="BH",
Cv111 = lfcShrink(dds, contrast=c("group","111","Control"), res=Cv111)
```

```
using 'normal' for LFC shrinkage, the Normal prior from Love et al (201
4).
additional priors are available via the 'type' argument, see ?lfcShrink f
or details
```

- Summary

In [25]:
```
summary(Cv111)
```

```
out of 246300 with nonzero total read count
adjusted p-value < 0.001
LFC > 0 (up)       : 378, 0.15%
LFC < 0 (down)     : 1173, 0.48%
outliers [1]       : 2543, 1%
low counts [2]     : 133688, 54%
(mean count < 5)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

- Contrast 653 vs 111

In [26]:
```
i653v111 = results(dds, contrast=c("group","653","111"),
                   independentFiltering=TRUE, alpha=0.001, pAdjustMethod="BH",
i653v111 = lfcShrink(dds, contrast=c("group","653","111"), res=i653v111)
```

```
using 'normal' for LFC shrinkage, the Normal prior from Love et al (201
4).
additional priors are available via the 'type' argument, see ?lfcShrink f
or details
```

- Summary

In [27]:
```
summary(i653v111)
```

```
out of 246300 with nonzero total read count
adjusted p-value < 0.001
LFC > 0 (up)       : 84, 0.034%
LFC < 0 (down)     : 14, 0.0057%
outliers [1]       : 2543, 1%
low counts [2]     : 152475, 62%
(mean count < 8)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

- **Subset gene with > or < log 2 fold change and q value < 0.05 (p value post FDR adjustment)**

- First define the cutoffs for log2 fold differences and the q value

In [67]:
```
log2cutoff = 2
qvaluecutoff = 0.001
```

- Concatenate results that are differentially expressed (>log2) and adjusted p-values < $q = 0.05$

In [68]:
```
diffXGenes <- unique(c(
  rownames(subset(Cv653, padj<=qvaluecutoff & abs(log2FoldChange)>=log2cutc
  rownames(subset(Cv111, padj<=qvaluecutoff & abs(log2FoldChange)>=log2cutc
  rownames(subset(i653v111, padj<=qvaluecutoff & abs(log2FoldChange)>=log2c
```

- Build assay object

In [69]:
```
heat <- assay(rld)[diffXGenes,]
```

- **Isolate genes for each comparison and sort by the log2 fold change estimates**

- Down-regulated genes

In [41]:
```
resSig_Cv653 = subset(Cv653, padj < 0.001)# control vs 653
```

In [42]:
```
head(resSig_Cv653[ order(resSig_Cv653$log2FoldChange), ])
```

```
log2 fold change (MAP): group 653 vs Control
Wald test p-value: group 653 vs Control
DataFrame with 6 rows and 6 columns
                           baseMean    log2FoldChange             lfcSE
                          <numeric>         <numeric>         <numeric>
Cluster-67248.65881  202.951197945512 -6.28646721243566 0.321043460493498
Cluster-67248.4354   114.361793332159 -6.10139058349401 0.324703217984243
Cluster-6227.0         28.413370682172 -3.81566588608497 0.364050345593477
Cluster-67248.121974 90.6587676924926 -3.64539680421656 0.338107941310253
Cluster-67248.98511  19.2627686051595 -3.18336169460425 0.366931856249029
Cluster-7595.0       18.0599056187062 -3.13211673429244 0.366926109301981
                                 stat               pvalue
                            <numeric>            <numeric>
Cluster-67248.65881  -11.6826157068244 1.56407165592058e-31
Cluster-67248.4354    -9.22015418935383 2.96673297884751e-20
Cluster-6227.0        -7.10607300633044 1.19391276197536e-12
Cluster-67248.121974  -7.68644909581806  1.5127483451757e-14
Cluster-67248.98511   -6.4482580684916 1.13142991747744e-10
Cluster-7595.0        -6.38722903844587 1.68918547510571e-10
                                   padj
                              <numeric>
Cluster-67248.65881   1.6475305194805e-26
Cluster-67248.4354    1.04167928353294e-15
Cluster-6227.0        1.04801662246197e-08
Cluster-67248.121974  2.65578099479047e-10
Cluster-67248.98511   4.25643934955012e-07
Cluster-7595.0        5.70727227013363e-07
```

In [51]:
```
resSig_Cv111 = subset(Cv111, padj < 0.001)# control vs 111
```

In [52]:
```
head(resSig_Cv111[ order(resSig_Cv111$log2FoldChange), ])
```

```
log2 fold change (MAP): group 111 vs Control
Wald test p-value: group 111 vs Control
DataFrame with 6 rows and 6 columns
                           baseMean    log2FoldChange             lfcSE
                          <numeric>         <numeric>         <numeric>
Cluster-67248.65881 202.951197945512 -6.55892034953777 0.326410632489976
Cluster-67248.4354  114.361793332159 -5.86374025346785 0.316705178166855
Cluster-6227.0       28.413370682172 -3.76118027552175 0.363723663894864
Cluster-67248.50623 55.2010260797926 -3.45561826424811 0.360606210708655
Cluster-67248.98511 19.2627686051595 -3.19922469856714 0.367038778951685
Cluster-7595.0      18.0599056187062 -3.07729077888003 0.367181471384888
                                stat               pvalue
padj
                           <numeric>            <numeric>            <nu
meric>
Cluster-67248.65881 -9.87466391769537 5.36123751216411e-23 3.105821324875
74e-19
Cluster-67248.4354  -10.8839277483815 1.37507301949253e-27 1.681699024250
26e-23
Cluster-6227.0      -6.39886290937604 1.56538307394028e-10 7.491310850675
34e-08
Cluster-67248.50623  -7.5734168754873 3.63533356619918e-14 5.065032029088
33e-11
Cluster-67248.98511 -6.49184279953414 8.47927119486211e-11 4.423245977001
32e-08
Cluster-7595.0      -5.70215111106775 1.18304878101351e-08 2.925388058689
15e-06
```

In [53]:
```
resSig_653v111 = subset(i653v111, padj < 0.001)# 653 vs 111
```

In [54]:
```
head(resSig_653v111[ order(resSig_653v111$log2FoldChange), ])
```

```
log2 fold change (MAP): group 653 vs 111
Wald test p-value: group 653 vs 111
DataFrame with 6 rows and 6 columns
                           baseMean    log2FoldChange             lfcSE
                          <numeric>         <numeric>         <numeric>
Cluster-67248.121974 90.6587676924926  -3.9589940802961 0.336368110261914
Cluster-67248.112206 18.5626354004593 -2.69648250849936 0.366845367519375
Cluster-67248.76854  24.3571393865261 -2.53083331116833 0.361370461597882
Cluster-67248.149503 27.8249871261726 -2.38866086700844 0.364384606107117
Cluster-67248.75344  15.0449546991255 -2.15398576339566 0.364284669558007
Cluster-67248.148461 9.72151122225922 -1.81687736592331 0.345862164163721
                               stat               pvalue
                          <numeric>            <numeric>
Cluster-67248.121974 -7.99585298827209 1.28679849746251e-15
Cluster-67248.112206 -5.50382972112057 3.71628503530835e-08
Cluster-67248.76854  -6.18707077234617 6.12924522094512e-10
Cluster-67248.149503  -6.2069334740323 5.40284555384753e-10
Cluster-67248.75344  -5.56746651963085 2.58469719742434e-08
Cluster-67248.148461 -4.98768053367799 6.11085072523991e-07
                                 padj
                            <numeric>
Cluster-67248.121974 1.46826925556716e-11
Cluster-67248.112206 6.65156726652974e-05
Cluster-67248.76854  2.79744881129156e-06
Cluster-67248.149503 2.59569762024374e-06
Cluster-67248.75344  4.91534019948517e-05
Cluster-67248.148461 0.000633875768069715
```

- **Up-regulated genes**

In [55]:
```
head(resSig_Cv653[ order(resSig_Cv653$log2FoldChange, decreasing = TRUE), ]
```

```
log2 fold change (MAP): group 653 vs Control
Wald test p-value: group 653 vs Control
DataFrame with 6 rows and 6 columns
                           baseMean   log2FoldChange             lfcSE
                          <numeric>        <numeric>         <numeric>
Cluster-67248.41609  124.279390389592 3.52725893385813 0.345570734128572
Cluster-67248.84245  40.9703792977569  3.4955170912277 0.366553535763028
Cluster-67248.13909   121.78368924237 3.42970727154396 0.299340298106147
Cluster-67248.142094 20.2956372731499  2.7908748658698 0.366388606928481
Cluster-71973.0      21.3301738242477 2.29032146110244 0.365199165612169
Cluster-67248.115536 79.6526611339483   2.288392884457 0.359349809555629
                                stat               pvalue
padj
                           <numeric>            <numeric>            <nu
meric>
Cluster-67248.41609  7.64229899956177 2.13376951603116e-14 3.210896367723
69e-10
Cluster-67248.84245  6.77707740483589 1.22631195264817e-11 7.176377546897
06e-08
Cluster-67248.13909  10.0175098153321 1.27679313466229e-23 6.724614081639
37e-19
Cluster-67248.142094  5.1099591792728 3.22228426089691e-07 0.000265173855
395185
Cluster-71973.0      5.68570176100883  1.3027657593149e-08 2.213357000374
11e-05
Cluster-67248.115536 6.60449100950853  3.9888541638035e-11 2.000809248563
83e-07
```

In [56]:
```
head(resSig_Cv111[ order(resSig_Cv111$log2FoldChange, decreasing = TRUE), ]
```

```
log2 fold change (MAP): group 111 vs Control
Wald test p-value: group 111 vs Control
DataFrame with 6 rows and 6 columns
                           baseMean   log2FoldChange             lfcSE
                          <numeric>        <numeric>         <numeric>
Cluster-67248.41609  124.279390389592 3.38418634239652 0.345620815354031
Cluster-67248.13909   121.78368924237 3.38307889682002 0.299295325657117
Cluster-67248.155958 51.2696932074041 2.48054331596675 0.367824731695813
Cluster-67248.112206 18.5626354004593 2.46892943178247 0.365128892872883
Cluster-67248.115536 79.6526611339483 2.43904926524883 0.359223108777017
Cluster-67248.76854  24.3571393865261 2.43072270277489 0.358678709538164
                                 stat               pvalue
padj
                            <numeric>            <numeric>            <nu
meric>
Cluster-67248.41609    7.496106389997 6.57414375079722e-14 8.824505225689
02e-11
Cluster-67248.13909  9.90931903031448  3.7922199806499e-23 2.608786631563
46e-19
Cluster-67248.155958 6.54049508941158 6.13155106650027e-11  3.29216436262
74e-08
Cluster-67248.112206 4.80749394072171 1.52834049779145e-06 0.000170438612
210139
Cluster-67248.115536 6.88865925610933 5.63206601016873e-12 4.132772491155
08e-09
Cluster-67248.76854  5.70729105119387 1.14788402827629e-08 2.858516902903
69e-06
```

In [57]:
```
head(resSig_653v111[ order(resSig_653v111$log2FoldChange, decreasing = TRUE
```

```
log2 fold change (MAP): group 653 vs 111
Wald test p-value: group 653 vs 111
DataFrame with 6 rows and 6 columns
                           baseMean   log2FoldChange             lfcSE
                          <numeric>        <numeric>         <numeric>
Cluster-67248.84245  40.9703792977569 3.70463793051371  0.36669338917001
Cluster-67248.142094 20.2956372731499 3.14473703586671 0.367133239478846
Cluster-67248.87571  23.3408647625934 2.36391076747395 0.364787539610653
Cluster-67248.132953 24.1723390703805 2.25356877582255 0.357912134362856
Cluster-67248.88523  30.3263718816141 2.03338159861437 0.362475626984758
Cluster-67248.50623  55.2010260797926 1.97704579017896 0.361559396776706
                                stat               pvalue
padj
                           <numeric>            <numeric>            <nu
meric>
Cluster-67248.84245    7.342528006312  2.0959664603159e-13 1.913240104305
56e-09
Cluster-67248.142094   6.552319395312 5.66502129690468e-11 2.872858189022
52e-07
Cluster-67248.87571  6.10019355209696 1.05940104359915e-09 4.029343585909
08e-06
Cluster-67248.132953 5.80984050003217 6.25323892095124e-09 1.585578208839
64e-05
Cluster-67248.88523  5.64312893998592 1.66987190358004e-08 3.464301070513
49e-05
Cluster-67248.50623  5.93363419791334 2.96301598160793e-09 9.015667494437
84e-06
```

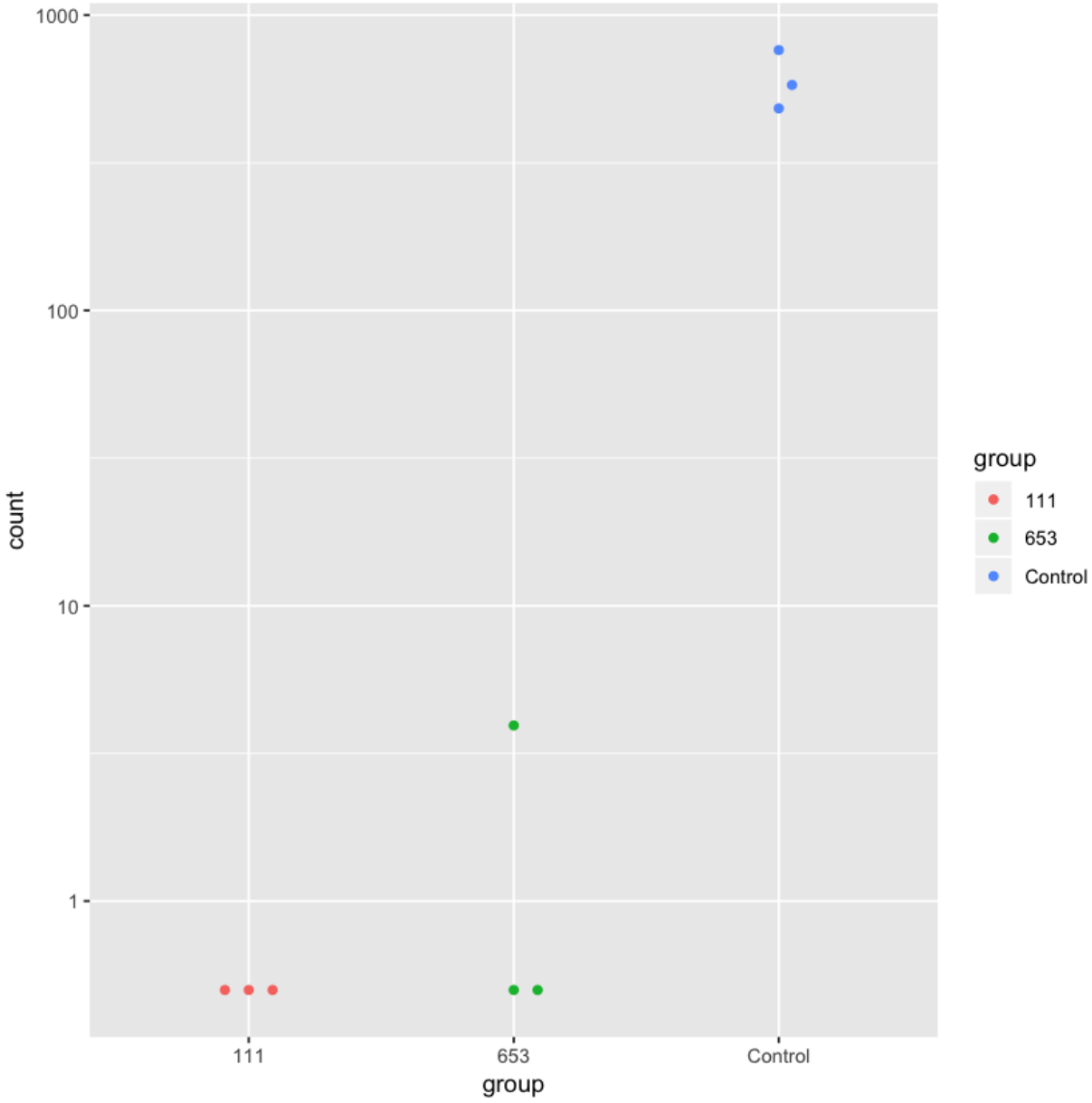
Visualization

Back to Table of Contents

- **Visualize normalized counts for single gene among treatments**

- With ggplot2

In [71]:
```
geneCounts <- plotCounts(dds, gene = topGene, intgroup = c("group"),
                         returnData = TRUE)
ggplot(geneCounts, aes(x = group, y = count, color = group)) +
  scale_y_log10() +  geom_beeswarm(cex = 3)
```



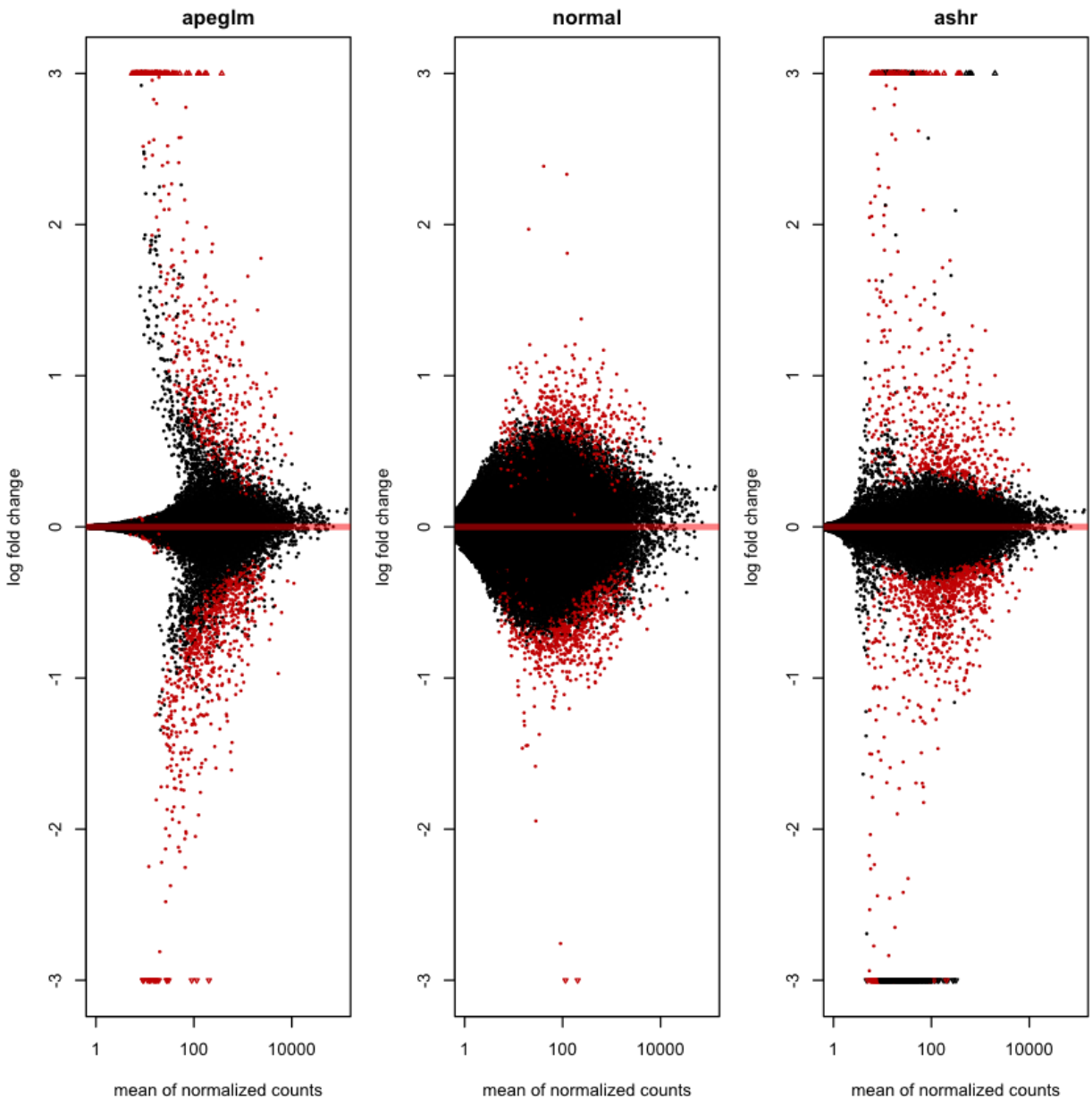
MA plot

- Control vs. 653

In [75]:
```
resLFC <- lfcShrink(dds, coef=3, type="apeglm")
resNorm <- lfcShrink(dds, coef=3, type="normal")
resAsh <- lfcShrink(dds, coef=3, type="ashr")
```

```
using 'apeglm' for LFC shrinkage. If used in published research, please c
ite:
    Zhu, A., Ibrahim, J.G., Love, M.I. (2018) Heavy-tailed prior distribu
tions for
    sequence count data: removing the noise and preserving large differen
ces.
    Bioinformatics. https://doi.org/10.1093/bioinformatics/bty895 (http
s://doi.org/10.1093/bioinformatics/bty895)
using 'normal' for LFC shrinkage, the Normal prior from Love et al (201
4).
additional priors are available via the 'type' argument, see ?lfcShrink f
or details
using 'ashr' for LFC shrinkage. If used in published research, please cit
e:
    Stephens, M. (2016) False discovery rates: a new deal. Biostatistics,
18:2.
    https://doi.org/10.1093/biostatistics/kxw041 (https://doi.org/10.109
3/biostatistics/kxw041)
```

In [76]:
```
par(mfrow=c(1,3), mar=c(4,4,2,1))
xlim <- c(1,1e5); ylim <- c(-3,3)
DESeq2::plotMA(resLFC, xlim=xlim, ylim=ylim, main="apeglm", colNonSig = "bl
DESeq2::plotMA(resNorm, xlim=xlim, ylim=ylim, main="normal", colNonSig = "b
DESeq2::plotMA(resAsh, xlim=xlim, ylim=ylim, main="ashr", colNonSig = "blac
```

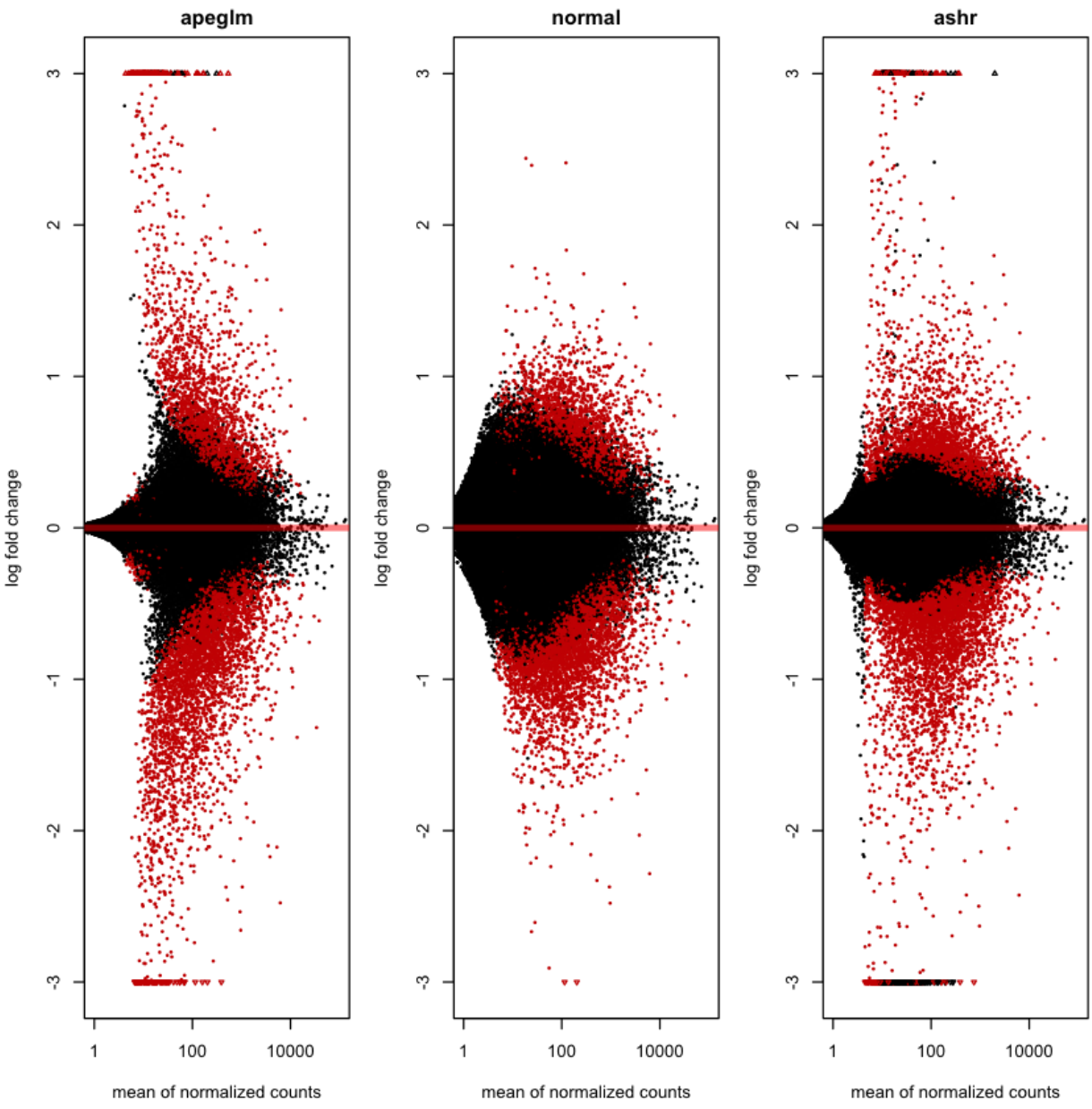


- 111 vs control

In [70]:
```
resLFC <- lfcShrink(dds, coef=2, type="apeglm")
resNorm <- lfcShrink(dds, coef=2, type="normal")
resAsh <- lfcShrink(dds, coef=2, type="ashr")
```

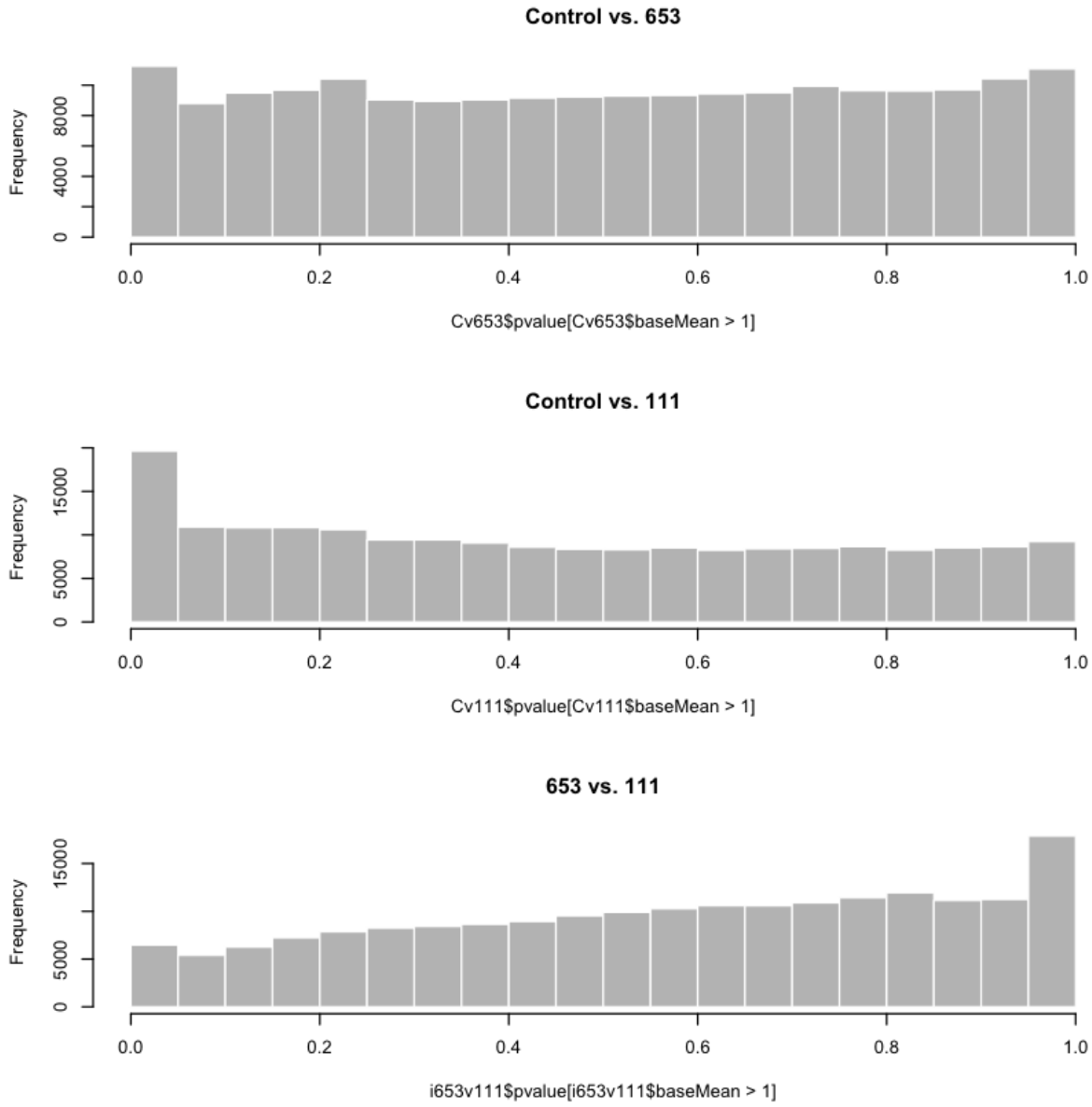
```
using 'apeglm' for LFC shrinkage. If used in published research, please c
ite:
    Zhu, A., Ibrahim, J.G., Love, M.I. (2018) Heavy-tailed prior distribu
tions for
    sequence count data: removing the noise and preserving large differen
ces.
    Bioinformatics. https://doi.org/10.1093/bioinformatics/bty895 (http
s://doi.org/10.1093/bioinformatics/bty895)
using 'normal' for LFC shrinkage, the Normal prior from Love et al (201
4).
additional priors are available via the 'type' argument, see ?lfcShrink f
or details
using 'ashr' for LFC shrinkage. If used in published research, please cit
e:
    Stephens, M. (2016) False discovery rates: a new deal. Biostatistics,
18:2.
    https://doi.org/10.1093/biostatistics/kxw041 (https://doi.org/10.109
3/biostatistics/kxw041)
```

In [72]:
```
par(mfrow=c(1,3), mar=c(4,4,2,1))
xlim <- c(1,1e5); ylim <- c(-3,3)
DESeq2::plotMA(resLFC, xlim=xlim, ylim=ylim, main="apeglm", colNonSig = "bl
DESeq2::plotMA(resNorm, xlim=xlim, ylim=ylim, main="normal", colNonSig = "b
DESeq2::plotMA(resAsh, xlim=xlim, ylim=ylim, main="ashr", colNonSig = "blac
```



Histograms of p-values

In [64]:
```
par(mfrow=c(3,1))
hist(Cv653$pvalue[Cv653$baseMean > 1], breaks = 0:20/20,
     col = "grey75", border = "white", main = "Control vs. 653")
hist(Cv111$pvalue[Cv111$baseMean > 1], breaks = 0:20/20,
     col = "grey75", border = "white", main = "Control vs. 111")
hist(i653v111$pvalue[i653v111$baseMean > 1], breaks = 0:20/20,
     col = "grey75", border = "white", main = "653 vs. 111")
```



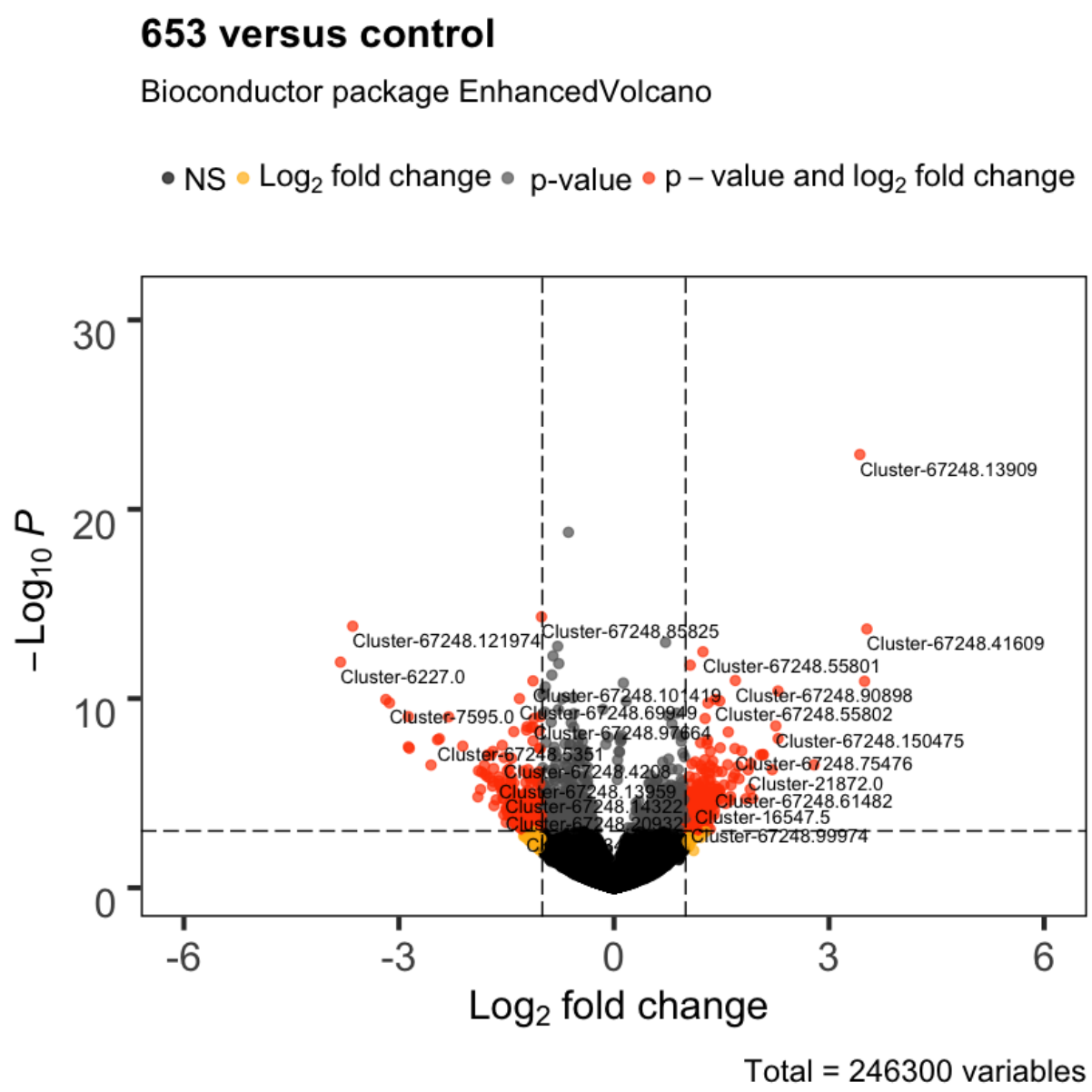
Volcano plots

- 653 vs control

In [34]:
```
Cv6=EnhancedVolcano(Cv653,
    lab = rownames(Cv653),
    x = 'log2FoldChange',
    y = 'pvalue',
    title = "653 versus control",
    legend=c("NS","Log2 fold-change","p-value",
    "p-value & Log2 fold-change"),
    legendPosition = "top",
    legendLabSize = 14,
    legendIconSize = 2.0,
    pCutoff = 0.001,
    FCcutoff = 1.0,
    transcriptPointSize = 1.75,
    transcriptLabSize = 3.0,
    colAlpha = 0.7,
    border = "full",
    gridlines.major = FALSE,
    gridlines.minor = FALSE,
    xlim = c(-6, 6),
    ylim = c(0, min(log10(Cv653$pvalue))),
    col=c("black", "darkgoldenrod1", "gray38", "orangered1"))

Cv6 + scale_color_manual(
  values=c(
    NS="black",
    FC="darkgoldenrod1",
    P="gray38",
    FC_P="orangered1"),
  labels=c(
    NS='NS',
    FC=expression(Log[2]~fold~change),
    P="p-value",
    FC_P=expression(p-value~and~log[2]~fold~change)))
```

```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



In [30]:
```
setwd("/Users/davidwheeler/Desktop/RESEARCH/Data/TRANSCRIPTOMICS/FIGURES")
```

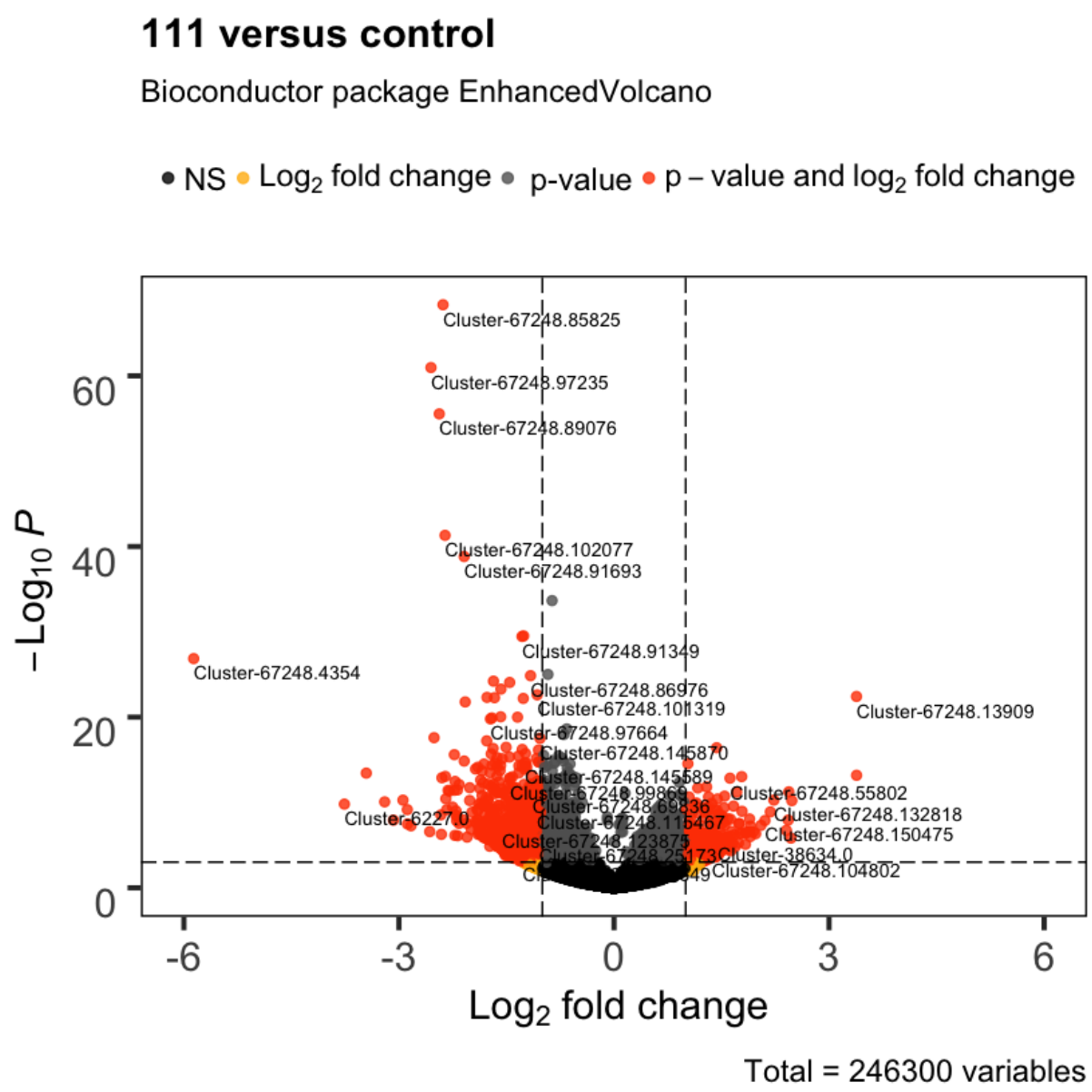
In [35]:
```
tiff("Mint_VP_Cv653.tiff", width=10, height=10, units='in', res=300)
Cv6
dev.off()
```

pdf: 2

- 111 vs control

In [36]:
```
Cv1=EnhancedVolcano(Cv111,
    lab = rownames(Cv111),
    x = 'log2FoldChange',
    y = 'pvalue',
    title = "111 versus control",
    legend=c("NS","Log2 fold-change","p-value",
    "p-value & Log2 fold-change"),
    legendPosition = "top",
    legendLabSize = 14,
    legendIconSize = 2.0,
    pCutoff = 0.001,
    FCcutoff = 1.0,
    transcriptPointSize = 1.75,
    transcriptLabSize = 3.0,
    colAlpha = 0.8,
    border = "full",
    gridlines.major = FALSE,
    gridlines.minor = FALSE,
    xlim = c(-6, 6),
    ylim = c(0, min(log10(Cv111$pvalue))),
    col=c("black", "darkgoldenrod1", "gray38", "orangered1"))
Cv1 + scale_color_manual(
  values=c(
    NS="black",
    FC="darkgoldenrod1",
    P="gray38",
    FC_P="orangered1"),
  labels=c(
    NS='NS',
    FC=expression(Log[2]~fold~change),
    P="p-value",
    FC_P=expression(p-value~and~log[2]~fold~change)))
```

```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



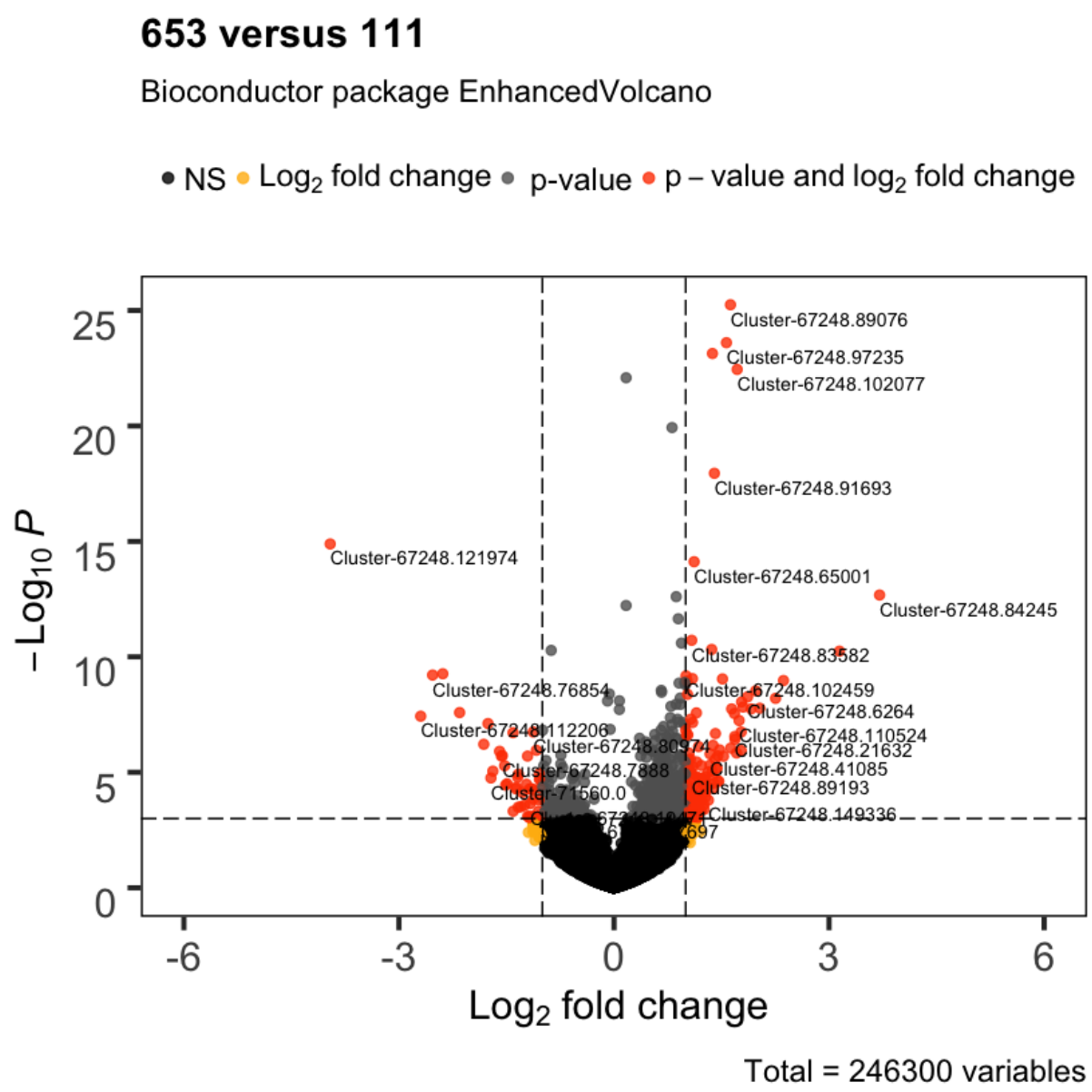
In [37]:
```
tiff("Mint_VP_Cv111.tiff", width=10, height=10, units='in', res=300)
Cv1
dev.off()
```

pdf: 2

- 653 vs. 111

In [38]:
```
i6vi1=EnhancedVolcano(i653v111,
    lab = rownames(i653v111),
    x = 'log2FoldChange',
    y = 'pvalue',
    title = "653 versus 111",
    legend=c("NS","Log2 fold-change","p-value",
    "p-value & Log2 fold-change"),
    legendPosition = "top",
    legendLabSize = 14,
    legendIconSize = 2.0,
    pCutoff = 0.001,
    FCcutoff = 1.0,
    transcriptPointSize = 1.75,
    transcriptLabSize = 3.0,
    colAlpha = 0.8,
    border = "full",
    gridlines.major = FALSE,
    gridlines.minor = FALSE,
    xlim = c(-6, 6),
    ylim = c(0, min(log10(i653v111$pvalue))),
    col=c("black", "darkgoldenrod1", "gray38", "orangered1"))
i6vi1 + scale_color_manual(
  values=c(
    NS="black",
    FC="darkgoldenrod1",
    P="gray38",
    FC_P="orangered1"),
  labels=c(
    NS='NS',
    FC=expression(Log[2]~fold~change),
    P="p-value",
    FC_P=expression(p-value~and~log[2]~fold~change)))
```

```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



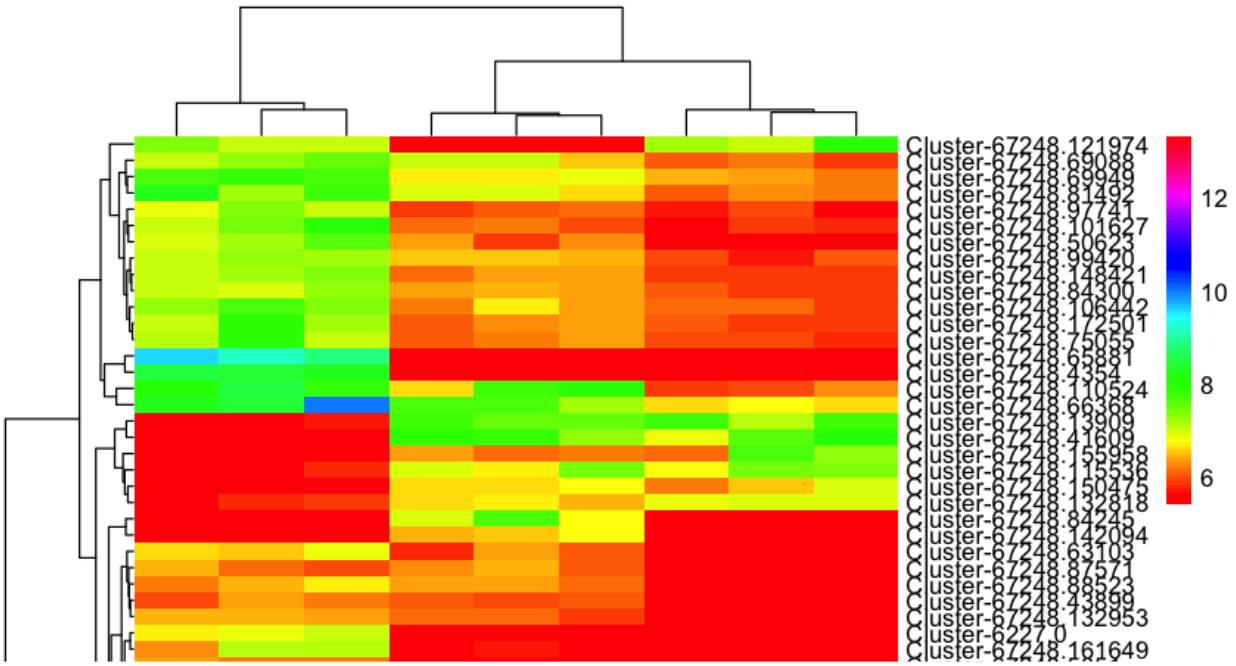
In [40]:
```
tiff("Mint_VP_653v111.tiff", width=10, height=10, units='in', res=300)
i6vi1
dev.off()
```

pdf: 2

Cluster Genes

In [31]:
```
# List with colors for each annotation
mat_colors <- list(group = brewer.pal(3, "RdYlBu"))
```

In [70]:
```
heat <- assay(vsd)[diffXGenes,]
Mint_DEGs=pheatmap(heat,
         color= rainbow(96),#,s=1,v=0.6,start=0.5,end=0.01),
         border_color = NA,
         show_colnames = TRUE,
         show_rownames = TRUE,
         labels_col=paste0(c("control","  "," ", "653","  "," ", "111","  "," "
         angle_col=0)
         #labels_col=paste0("bar", 1:10))
```



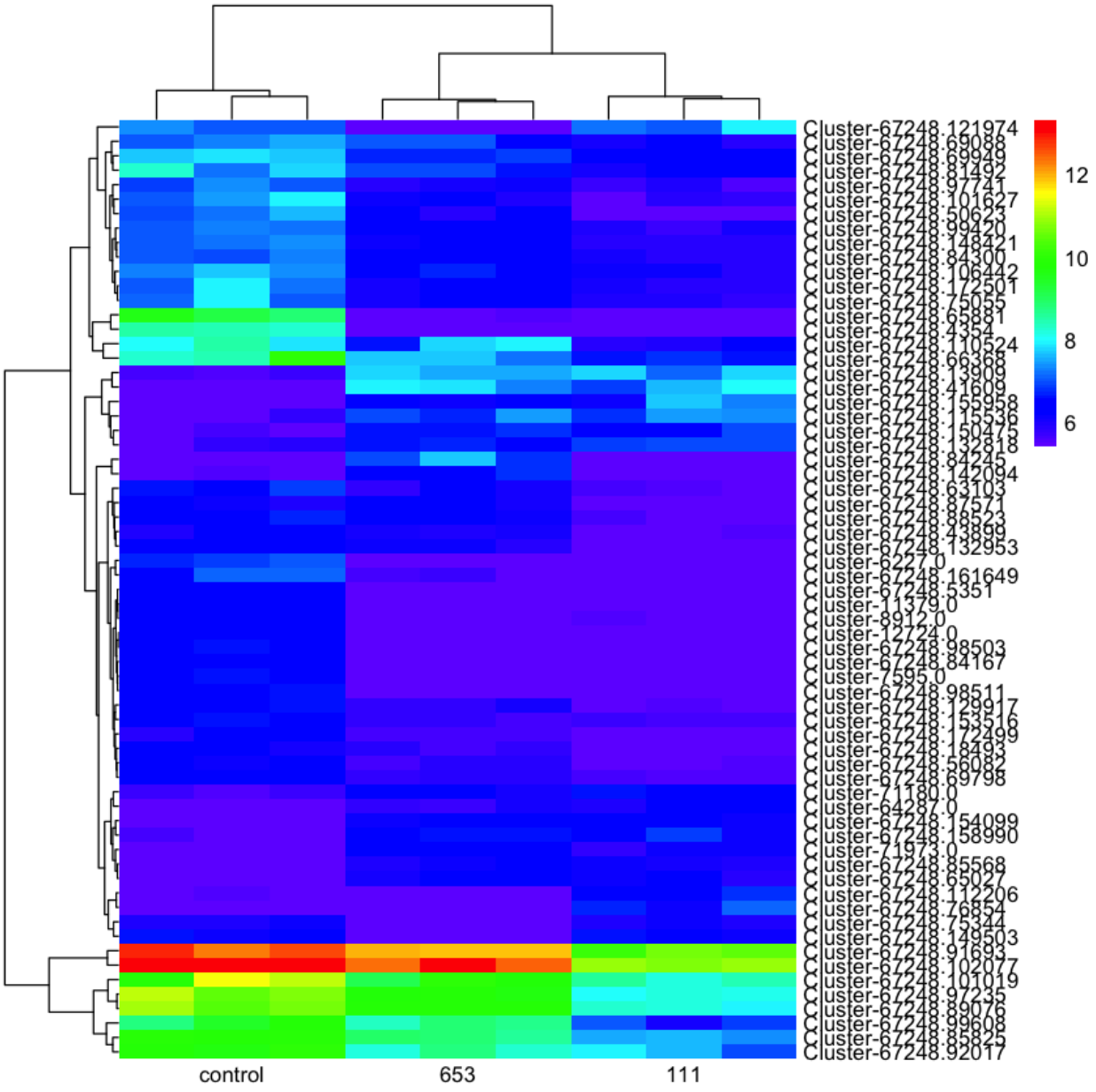
In [71]:
```
tiff("Mint_HeatMap_1.tiff", width = 5, height =10, units = 'in', res = 300)
Mint_DEGs
dev.off()
```

pdf: 2

In [91]:
```
getwd()
```

'/Users/davidwheeler/Desktop/RESEARCH/Data/TRANSCRIPTOMICS/DATA/R_FILES'

In [72]:
```
heat <- assay(vsd)[diffXGenes,]
Mint_DEGs=pheatmap(heat,
        color= rev(rainbow(96,start=0.0,end=0.74,alpha=1)),#,s=1,v=0.6,star
        border_color = NA,
        show_colnames = TRUE,
        show_rownames = TRUE,
        labels_col=paste0(c("control","  "," ", "653","  "," ", "111","  "," "
        angle_col=0)
        #labels_col=paste0("bar", 1:10))
```



In [73]:
```
tiff("Mint_HeatMap_2.tiff", width = 5, height =10, units = 'in', res = 300)
Mint_DEGs
dev.off()
```

pdf: 2

Venn Diagram

Control vs 653

In [43]:
```
resSig_Cv653 = subset(Cv653, padj < 0.001)
resSig_Cv653_fragments = row.names(resSig_Cv653)
```

Control vs 111

In [45]:
```
resSig_Cv111 = subset(Cv111, padj < 0.001)
resSig_Cv111_fragments = row.names(resSig_Cv111)
```

653 vs 111

In [46]:
```
resSig_653v111 = subset(i653v111, padj < 0.001)
resSig_653v111_fragments = row.names(resSig_653v111)
```

Build common dataframe

In [47]:
```
vdDF = c(resSig_Cv653_fragments,
         resSig_Cv111_fragments,
        resSig_653v111_fragments)
```

Compare

In [48]:
```
resSig_Cv653_fragments.2 <- vdDF %in% resSig_Cv653_fragments
resSig_Cv111_fragments.2 <- vdDF %in% resSig_Cv111_fragments
resSig_653v111_fragments.2 <- vdDF %in% resSig_653v111_fragments
```

Compute venn diagram counts

In [51]:
```
counts = cbind(resSig_Cv653_fragments.2, resSig_Cv111_fragments.2, resSig_6
vdcounts = vennCounts(counts)
```

Plot

In [79]:
```
tiff("Mint_VD.tiff", width=5, height=5, units = 'in', res = 300)
vennDiagram(vdcounts,
            cex=1,
            lwd=2,
            names=c("Control vs 653","Control vs 111","653 vs 111"),
            circle.col = c("orangered2","skyblue3","orange1"))

dev.off()
```

pdf: 2

Export data

Back to Table of Contents

- Control vs. 653

In [ ]:
```
write.csv(as.data.frame(resSig_Cv653),
          file="Mentha_DEGs_Cv653.csv")
```

- Control vs. 111

In [ ]:
```
write.csv(as.data.frame(resSig_Cv111),
          file="Mentha_DEGs_Cv111.csv")
```

- 653 vs 111

In [ ]:
```
write.csv(as.data.frame(resSig_653v111),
          file="Mentha_DEGs_653v111.csv")
```

Graveyard

Back to Table of Contents

In [ ]:
```
# For each sample/row in the sample column
for (row in DF$Sample){
    # Split the sample by the underscore
    sample = strsplit(row,"_")
    # If sample contains S2,
    if (grepl("S2",sample)){
    }
}
```

Resources

Back to Table of Contents

- R kernel installation: https://irkernel.github.io/installation/ (https://irkernel.github.io/installation/)
- DESeq2 installation: https://anaconda.org/bioconda/bioconductor-deseq2 (https://anaconda.org/bioconda/bioconductor-deseq2)
- DESeq2 data curation: https://www.bioconductor.org/packages/devel/bioc/vignettes/DEFormats/inst/doc/DEFormats.ht (https://www.bioconductor.org/packages/devel/bioc/vignettes/DEFormats/inst/doc/DEFormats.h
- DESeq2 vignette: http://bioconductor.org/packages/devel/bioc/vignettes/DESeq2/inst/doc/DESeq2.html (http://bioconductor.org/packages/devel/bioc/vignettes/DESeq2/inst/doc/DESeq2.html) http://master.bioconductor.org/packages/release/workflows/vignettes/rnaseqGene/inst/doc/rnas (http://master.bioconductor.org/packages/release/workflows/vignettes/rnaseqGene/inst/doc/rnas
- Volcano plots: https://rdrr.io/github/kevinblighe/EnhancedVolcano/f/README.md (https://rdrr.io/github/kevinblighe/EnhancedVolcano/f/README.md)

- https://www.bioconductor.org/packages/release/bioc/vignettes/EnhancedVolcano/inst/doc/l
  (https://www.bioconductor.org/packages/release/bioc/vignettes/EnhancedVolcano/inst/doc/

Type *Markdown* and LaTeX: α^2

In [ ]: