

Preprocessing results

Sofia Illescas

October 21, 2022

```
setwd("~/Vascular_Disease/MG-04_Illumina_totalRNASeq/preprocess/")
#Importing data as summarized experiment object
se <- readRDS("~/Vascular_Disease/MG-04_Illumina_totalRNASeq/preprocess/patients_and_controls")

# Data exploration
metadata <- as.data.frame(colData(se))
fplot <- ggplot(as.data.frame(metadata))

# Quality assessment and normalization

## Preprocessing data
dge <- DGEList(counts = assays(se)$counts, genes = as.data.frame(mclocs(se)), group = se$Case.Control)

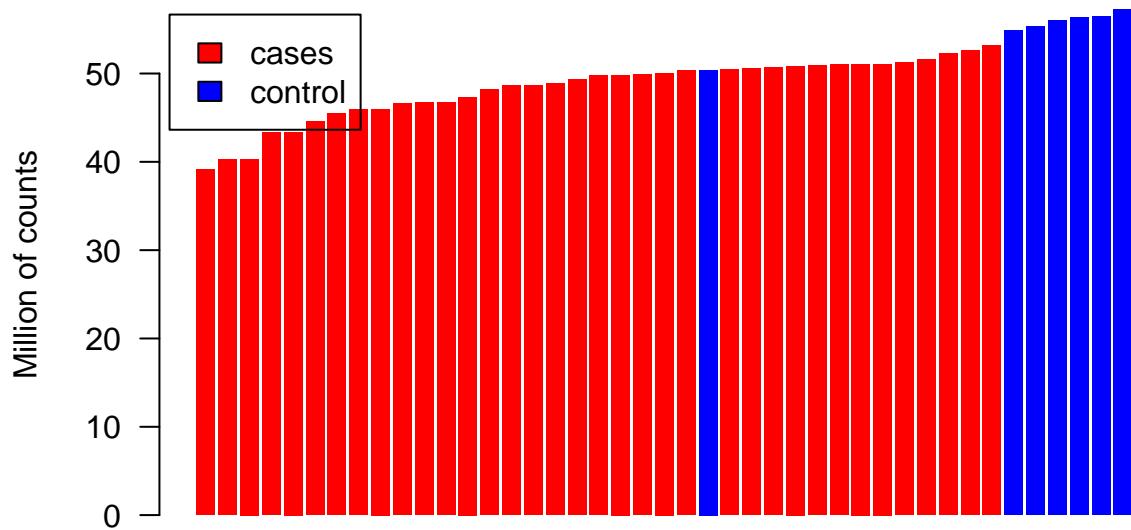
# Computing counts per million
assays(se)$logCPM <- cpm(dge, log=TRUE, prior.count=0.5)

## Examining the sequencing depth
ord <- order(dge$samples$lib.size)
dge$samples$lib.size[ord]

## [1] 39150701 40283271 40322833 43299317 43381491 44524742 45428208 45915328
## [9] 45975476 46609323 46698180 46754888 47342324 48208547 48616487 48682189
## [17] 48843691 49307670 49764726 49834319 49871002 50052172 50312942 50408628
## [25] 50460787 50603739 50690673 50857371 50898294 51054149 51077284 51086330
## [33] 51247663 51581366 52226271 52572181 53142216 54893959 55337749 55965002
## [41] 56326588 56415046 57256601

# Barplot patients/controls
barplot(dge$samples$lib.size[ord]/1e+06, las = 1, main="Barplot - library size", ylab = "Million of count"
legend("topleft", c("cases","control"), fill = c("red","blue"), inset = 0.01)
```

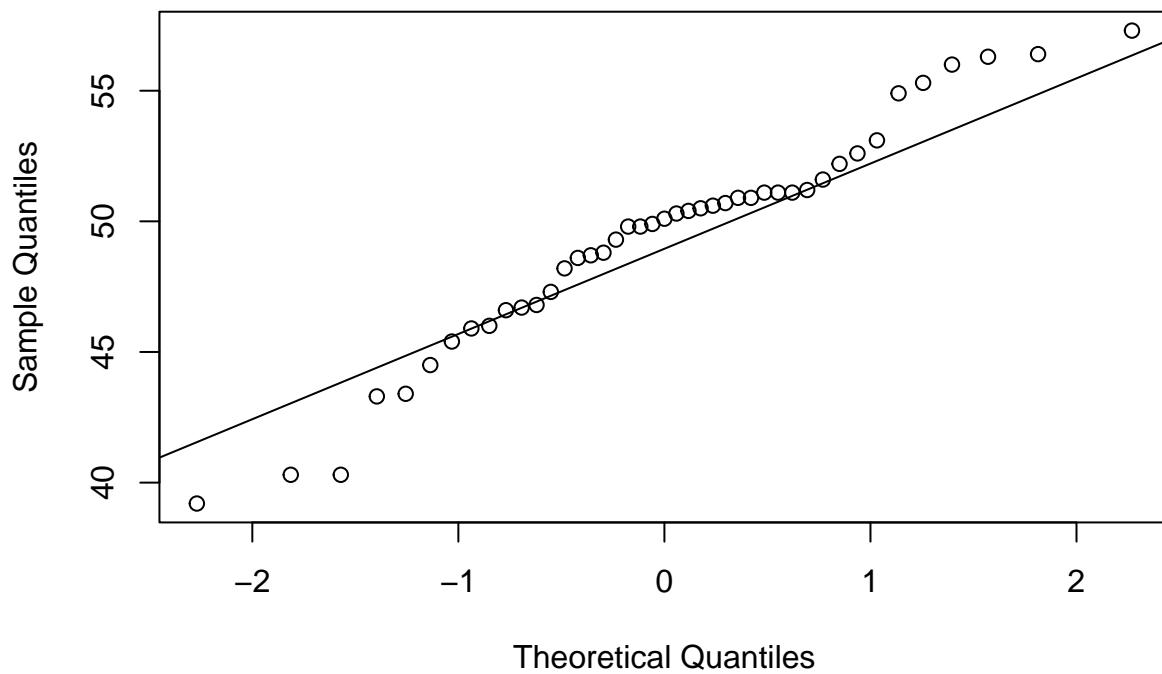
Barplot – library size



Samples

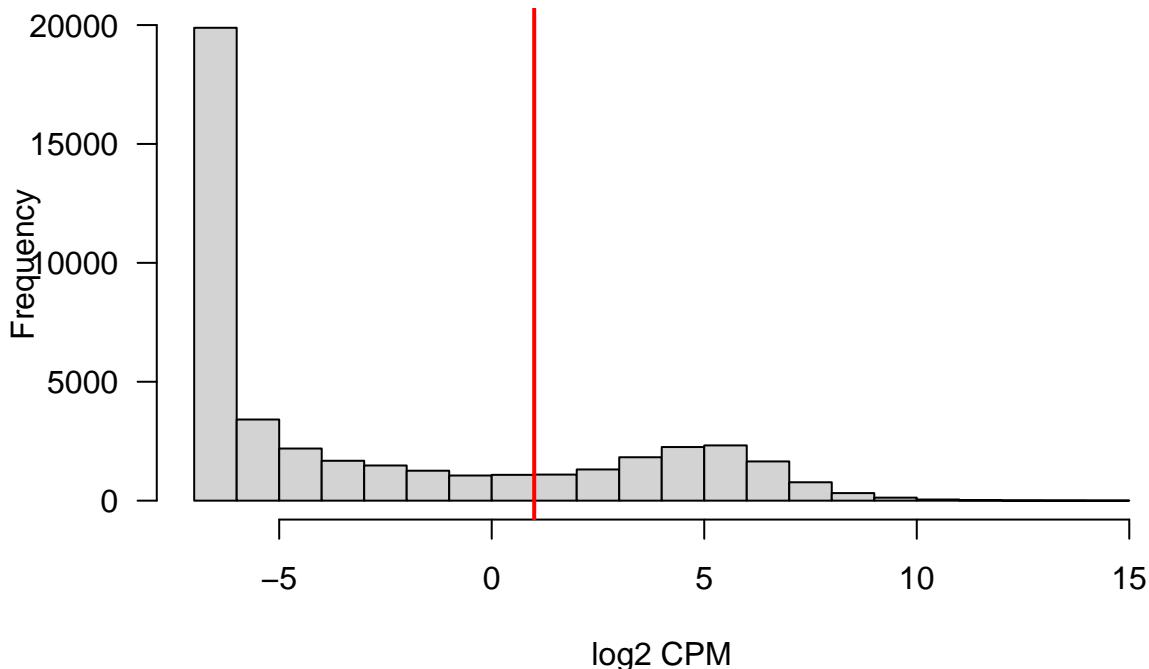
```
# QQplot
sampleddepth <- round(dge$sample$lib.size / 1e6, digits=1)
qqnorm(sampleddepth)
qqline(sampleddepth)
```

Normal Q–Q Plot



```
# Distribution of expression levels among genes
avgexp <- rowMeans(assays(se)$logCPM)
hist(avgexp, xlab="log2 CPM", main="Distribution of the genes average expression", las=1)
abline(v=1, col="red", lwd=2)
```

Distribution of the genes average expression



```
# Filtering lowly expressed genes
nsamples <- length(se$Process.ID)
sample_cutoff <- 0.2
nsamples_cutoff <- sample_cutoff*nsamples

logcpm_cutoff <- 1
mask <- rowSums(assays(se)$logCPM <= logcpm_cutoff) >= nsamples_cutoff

dim(se)

## [1] 43809     43

se.filt <- se[!mask, ]
dim(se.filt)

## [1] 10986     43

dge.filt <- dge[!mask, ]
dim(dge.filt)

## [1] 10986     43

kept_genes2b <- dim(se.filt)[1]

# Between sample normalization
dge.filt.norm <- calcNormFactors(dge.filt)
assays(se.filt)$logCPM.norm <- cpm(dge.filt.norm, log = TRUE, prior.count = 3, normalized.lib.sizes = T)
```

```

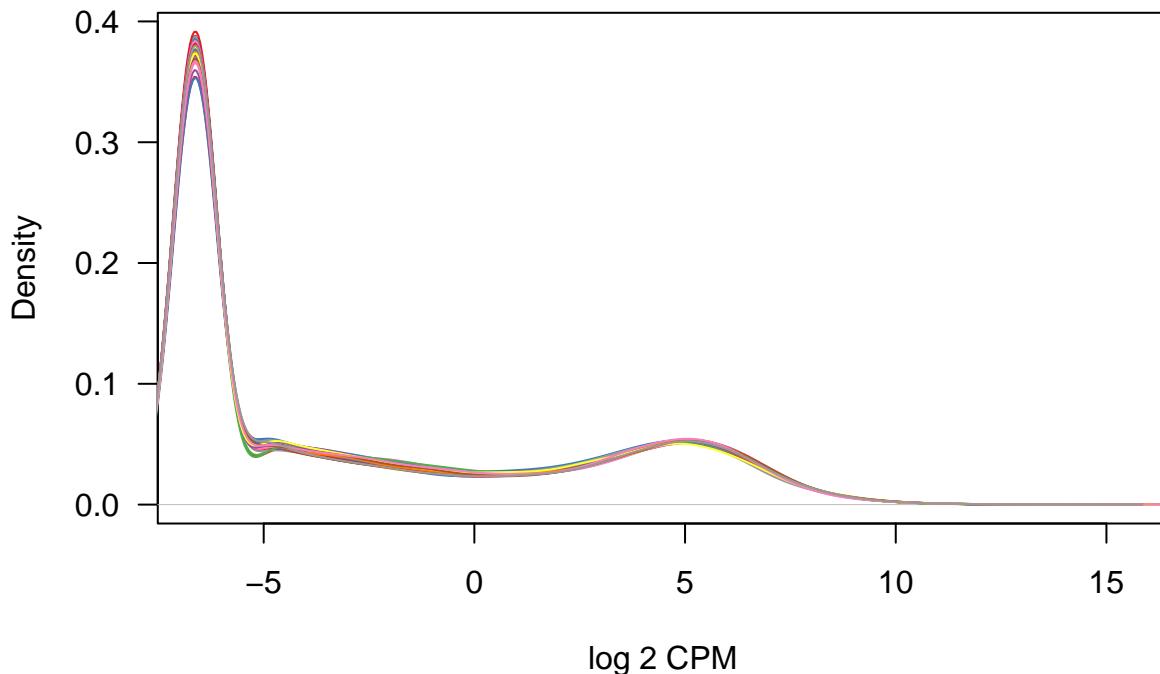
saveRDS(se.filt, "~/Vascular_Disease/MG-04_Illumina_totalRNASeq/preprocess/se_filt")

# Plots: in-sample normalized only, filtered, filtered and normalized between samples

multidensity(as.list(as.data.frame(assays(se[, se$Case.Control == "case"])$logCPM)),
             xlab="log 2 CPM", legend=NULL, main="Patient samples", las=1)

```

Patient samples

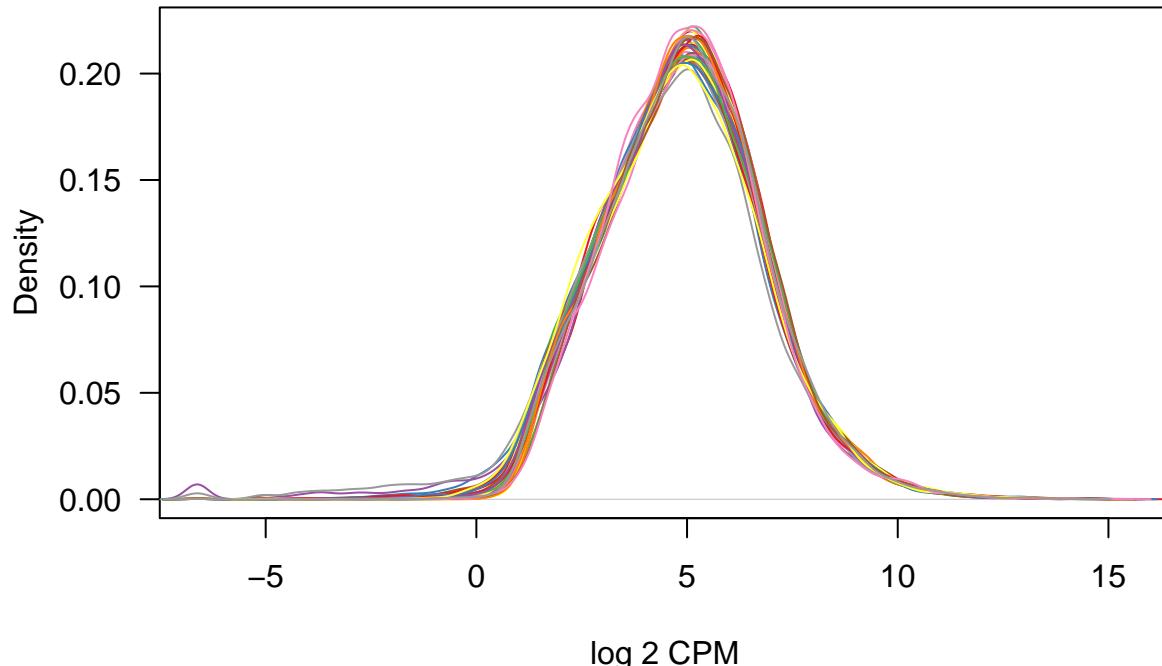


```

multidensity(as.list(as.data.frame(assays(se.filt[, se.filt$Case.Control == "case"])$logCPM)),
             xlab="log 2 CPM", legend=NULL, main="Patient samples", las=1)

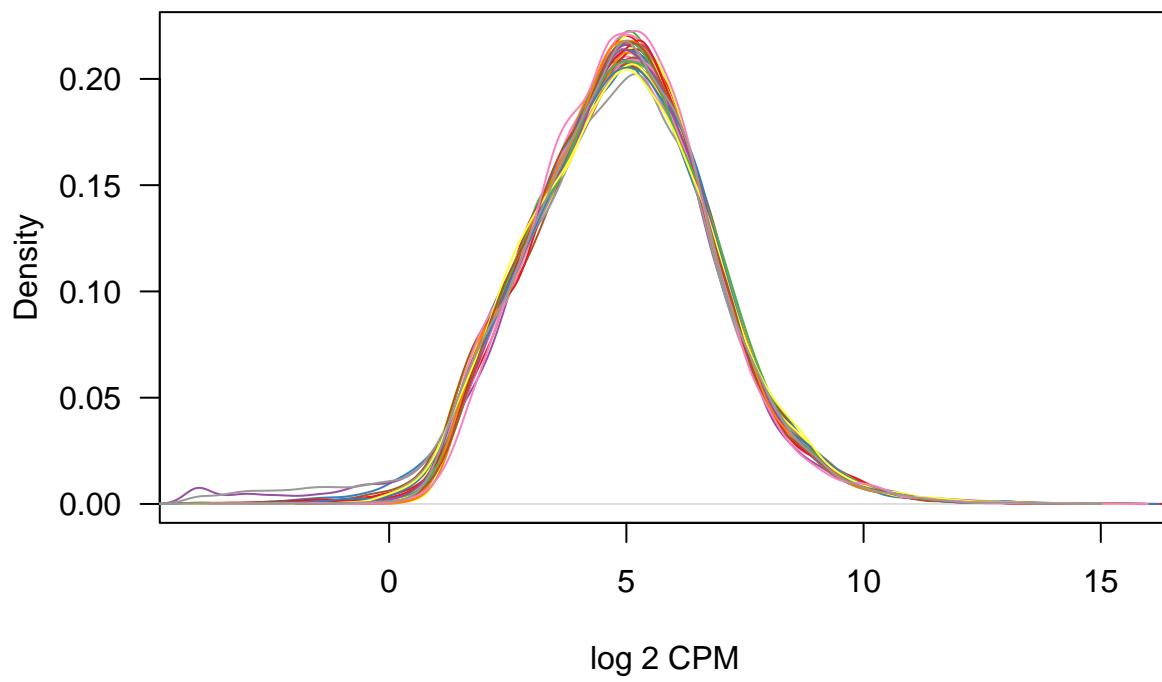
```

Patient samples



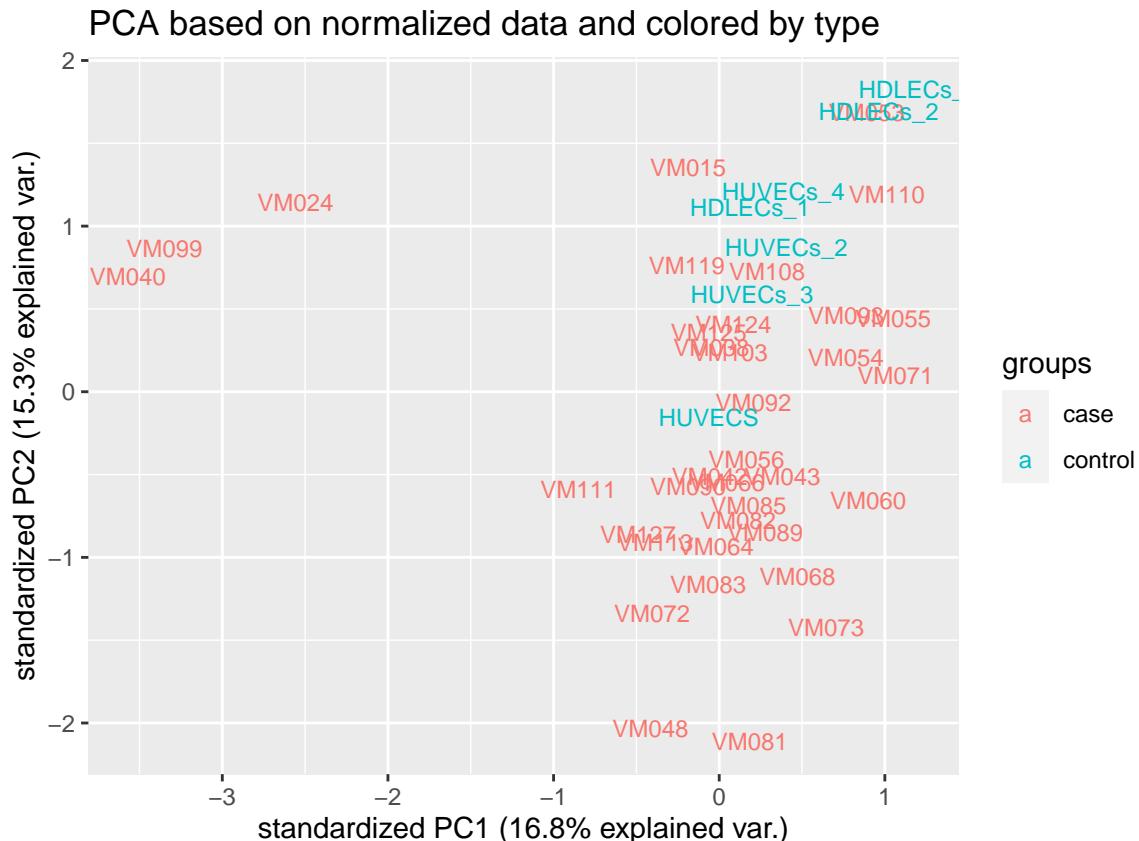
```
multidensity(as.list(as.data.frame(assays(se.filt[, se.filt$Case.Control == "case"])$logCPM.norm)),  
            xlab="log 2 CPM", legend=NULL, main="Patient samples", las=1)
```

Patient samples



```
# PCA based on normalized counts(normalized between samples)  
pca_norm_pre <- prcomp(as.data.frame(t(assays(se.filt)$logCPM.norm)) , center=TRUE, scale=TRUE)
```

```
ggbiplot(pca_norm_pre, var.axes=FALSE, groups=se.filt$Case.Control, labels = colnames(se.filt)) + ggtitle("PCA based on normalized data and colored by type")
```



```
##### DEAnalysis #####
```

```
##All controls vs all patients
```

```
# Without correcting for batch
```

```
se.filt$Case.Control <- relevel(factor(se.filt$Case.Control), ref="control")
mod <- model.matrix(~ se.filt$Case.Control, colData(se.filt))
mod0 <- model.matrix(~ 1, colData(se.filt))
pv <- f.pvalue(assays(se.filt)$logCPM.norm, mod, mod0)
sum(p.adjust(pv, method="fdr") < 0.05)
```

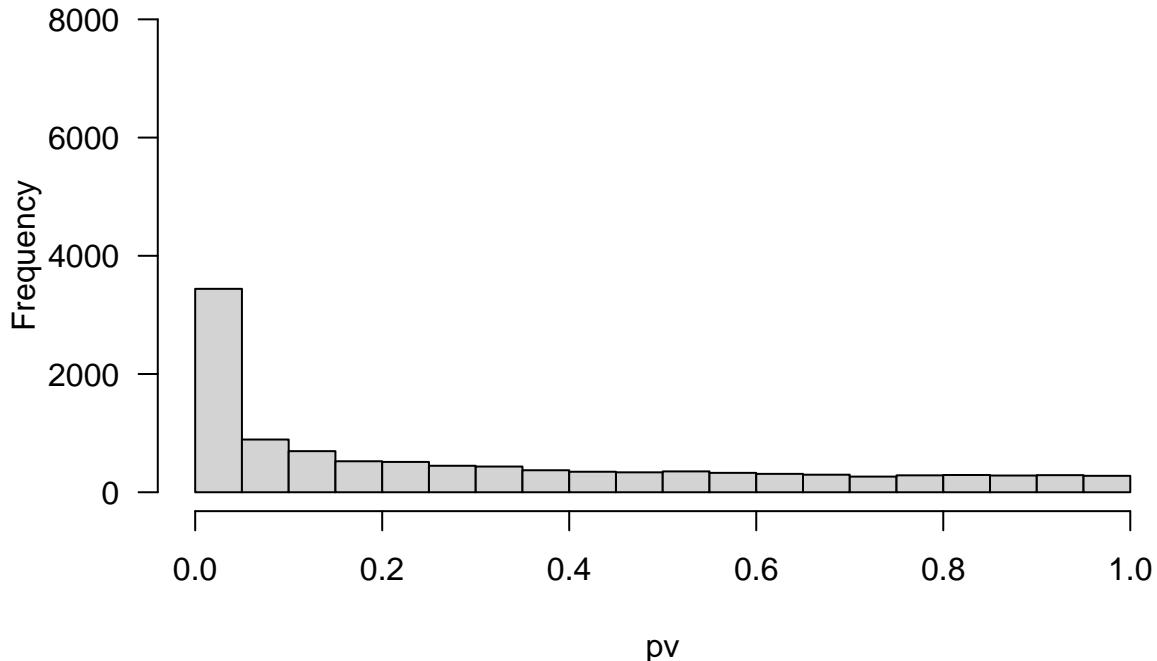
```
## [1] 1811
```

```
# 1811 DE genes, before mean-variance relationship
```

```
# Distribution of expected p-values
```

```
par(mfrow=c(1, 1))
hist(pv, main="Distribution of expected p-values", las=1, ylim = c(0, 8000))
```

Distribution of expected p-values

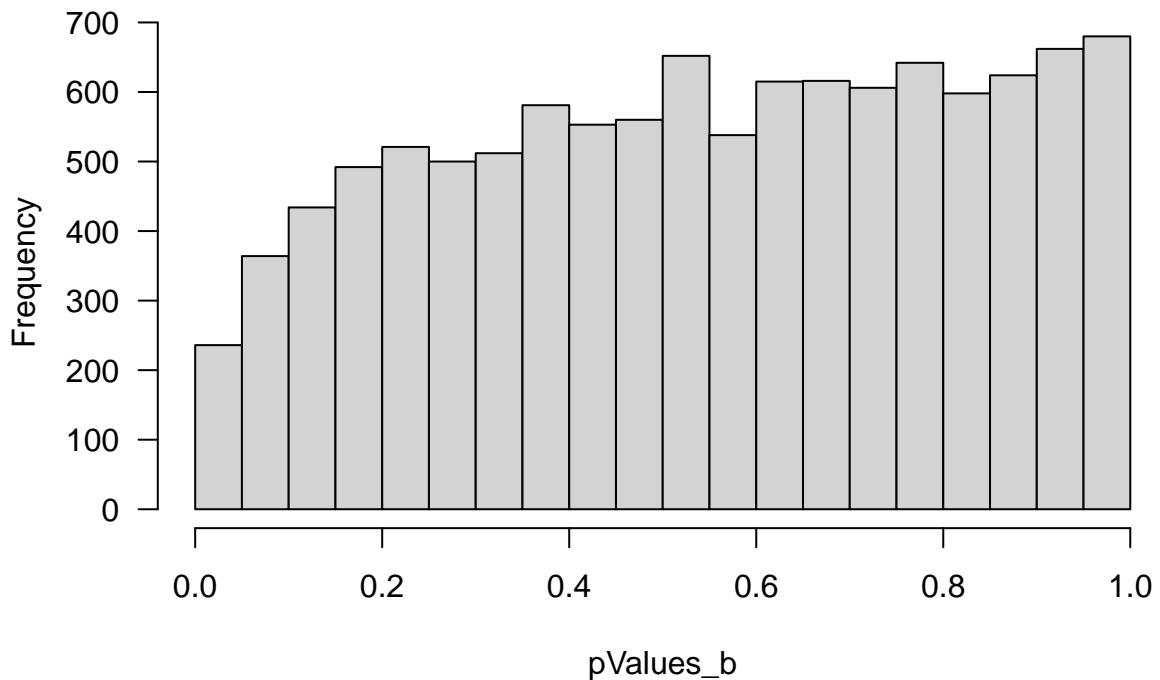


```
# Correcting for batch
se.filt$Batch <- c(1:43)
se.filt$Batch[1:37] <- "1"
se.filt$Batch[38:43] <- "2"
if (length(unique(se.filt$Batch)) > 1){
  mod_b <- model.matrix(~Case.Control + Batch, colData(se.filt))
  mod0_b <- model.matrix(~Batch, colData(se.filt))
  colnames(mod_b)
  pValues_b <- f.pvalue(assays(se.filt)$logCPM, mod_b, mod0_b)
  sum(p.adjust(pValues_b, method="fdr") < 0.05)
} else {
  print("There is only one series for this disease")
}

## [1] 0
#No DE genes when accounting for batch number

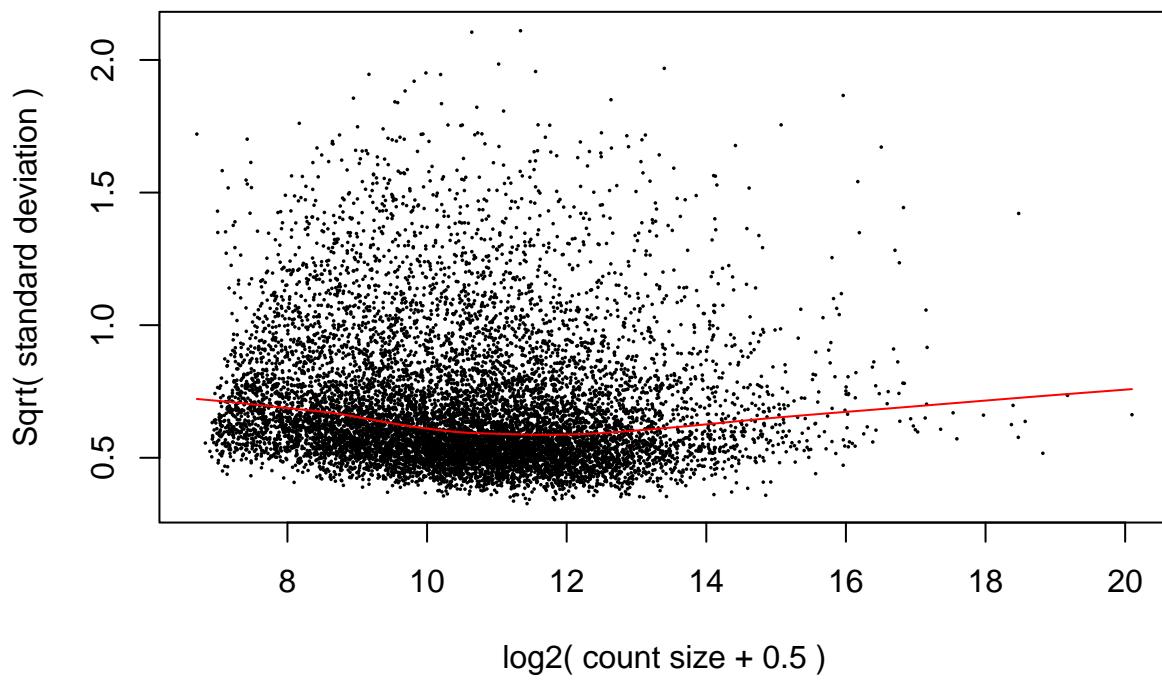
# Distribution of expected p-values after correcting for the series
if (length(unique(se.filt$Batch)) > 1){
  hist(pValues_b, main="Distribution of expected p-values after correcting for series", las=1)
```

Distribution of expected p-values after correcting for series



```
# Mean-variance relationship
FDRcutoff <- 0.05
v <- voom(dge.filt.norm, mod, plot = TRUE) # Voom is applied to the normalized and filtered DGEList obj
```

voom: Mean–variance trend



```

fit <- lmFit(v, mod)
fit<- eBayes(fit)
res <- decideTests(fit, p.value = FDRcutoff)
summary(res)

##          (Intercept) se.filt$Case.Controlcase
## Down              0                      1001
## NotSig            22                     9051
## Up               10964                   934

# 1935 DE genes, these are selected

# Add gene metadata (gene symbol) and fetch the table of results. Print first 15 DEgenes
rowRanges(se.filt)

## NULL

genesmd <- data.frame(symbol = rownames(res), stringsAsFactors = FALSE)
fit$genes <- genesmd
tt <- topTable(fit, coef = 2, n = Inf)
head(tt, 15)

##           symbol      logFC   AveExpr       t     P.Value
## LOC105374836 LOC105374836  2.3394926 5.724802  9.039808 1.606310e-11
## LOC105375683 LOC105375683  1.6095971 1.869564  8.318650 1.618410e-10
## SCAI           SCAI      -0.7659173 3.549599 -8.065791 3.684782e-10
## SCARNA2        SCARNA2    1.8270865 6.196742  7.792646 9.021646e-10
## SNORA48        SNORA48    2.3630555 3.697018  7.677666 1.317678e-09
## SCARNA28       SCARNA28   2.5187662 1.630722  7.614637 1.622522e-09
## SNORA73A       SNORA73A   1.7330189 8.115776  7.482392 2.513330e-09
## LOC112268052 LOC112268052  0.9468124 4.169657  7.418994 3.101394e-09
## PPWD1          PPWD1     -0.6499264 3.854227 -7.335389 4.094002e-09
## RAB5IF          RAB5IF    1.8920254 3.031794  7.312125 4.423229e-09
## LOC105372117 LOC105372117  0.9153063 2.339712  7.306684 4.503992e-09
## SNORA73B       SNORA73B   1.4978981 7.672883  7.149970 7.590567e-09
## SLC25A18       SLC25A18   0.8618914 3.304229  7.181775 6.826842e-09
## NDUFC2          NDUFC2    -0.7143143 4.291734 -7.068465 9.963308e-09
## LAMTOR1         LAMTOR1    0.6890794 6.742099  7.031489 1.127310e-08
##           adj.P.Val      B
## LOC105374836  1.764692e-07 15.974113
## LOC105375683  8.889926e-07 13.385155
## SCAI           1.349367e-06 13.030142
## SCARNA2        2.477795e-06 12.218570
## SNORA48        2.895203e-06 11.631226
## SCARNA28       2.970838e-06 11.316711
## SNORA73A       3.944492e-06 11.234981
## LOC112268052  4.258989e-06 10.981194
## PPWD1          4.498260e-06 10.762135
## RAB5IF          4.498260e-06 10.484709
## LOC105372117  4.498260e-06 10.464224
## SNORA73B       6.414613e-06 10.184737
## SLC25A18       6.249974e-06 10.167453
## NDUFC2          7.818350e-06 9.923502
## LAMTOR1         8.256420e-06 9.809158

```

```

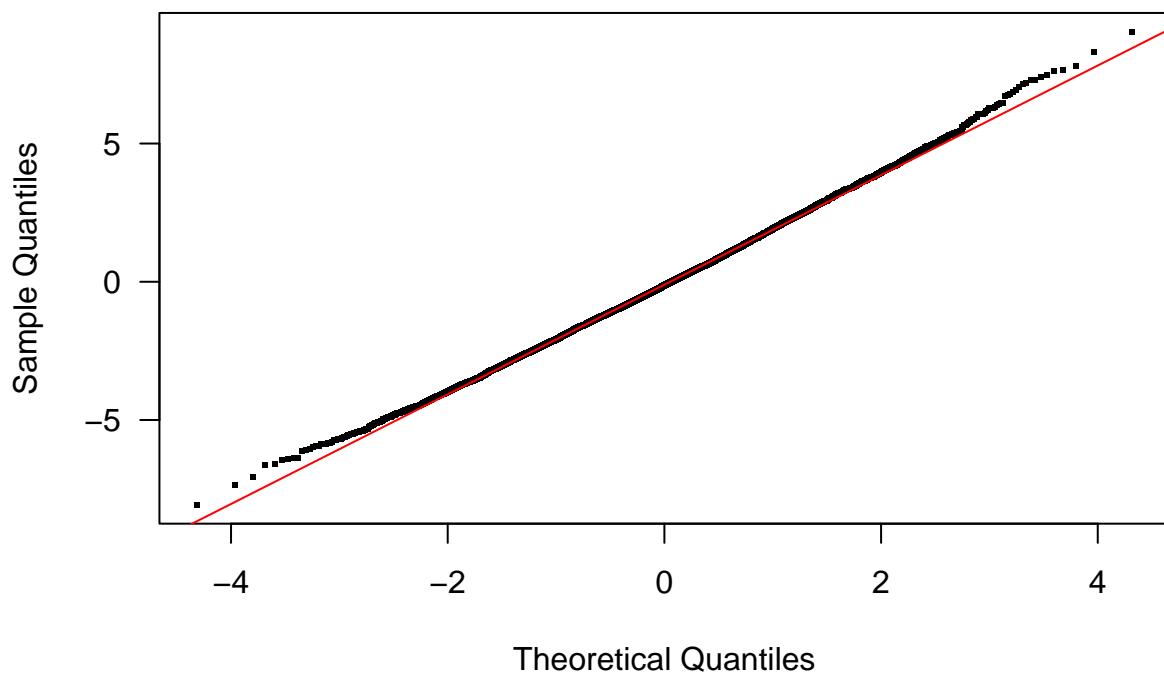
tt_sign <- tt[tt$adj.P.Val <= 0.05,]
head(tt_sign); nrow(tt_sign)

##          symbol      logFC AveExpr         t     P.Value
## LOC105374836 LOC105374836  2.3394926 5.724802 9.039808 1.606310e-11
## LOC105375683 LOC105375683  1.6095971 1.869564 8.318650 1.618410e-10
## SCAI           SCAI      -0.7659173 3.549599 -8.065791 3.684782e-10
## SCARNA2        SCARNA2   1.8270865 6.196742 7.792646 9.021646e-10
## SNORA48        SNORA48   2.3630555 3.697018 7.677666 1.317678e-09
## SCARNA28       SCARNA28  2.5187662 1.630722 7.614637 1.622522e-09
##          adj.P.Val      B
## LOC105374836 1.764692e-07 15.97411
## LOC105375683 8.889926e-07 13.38516
## SCAI           1.349367e-06 13.03014
## SCARNA2        2.477795e-06 12.21857
## SNORA48        2.895203e-06 11.63123
## SCARNA28       2.970838e-06 11.31671

## [1] 1935
sum(tt$adj.P.Val < 0.05)

## [1] 1935
genes <- (tt$symbol[(tt$adj.P.Val < 0.05)])
genes_or <- genes[order(genes)]

# Q-Q plot
{qqt(fit$t[, 2], df = fit$df.prior + fit$df.residual, main = "", pch = ".", cex = 3, las = 1)
 qqline(fit$t[, 2], col = "red")}
```



```

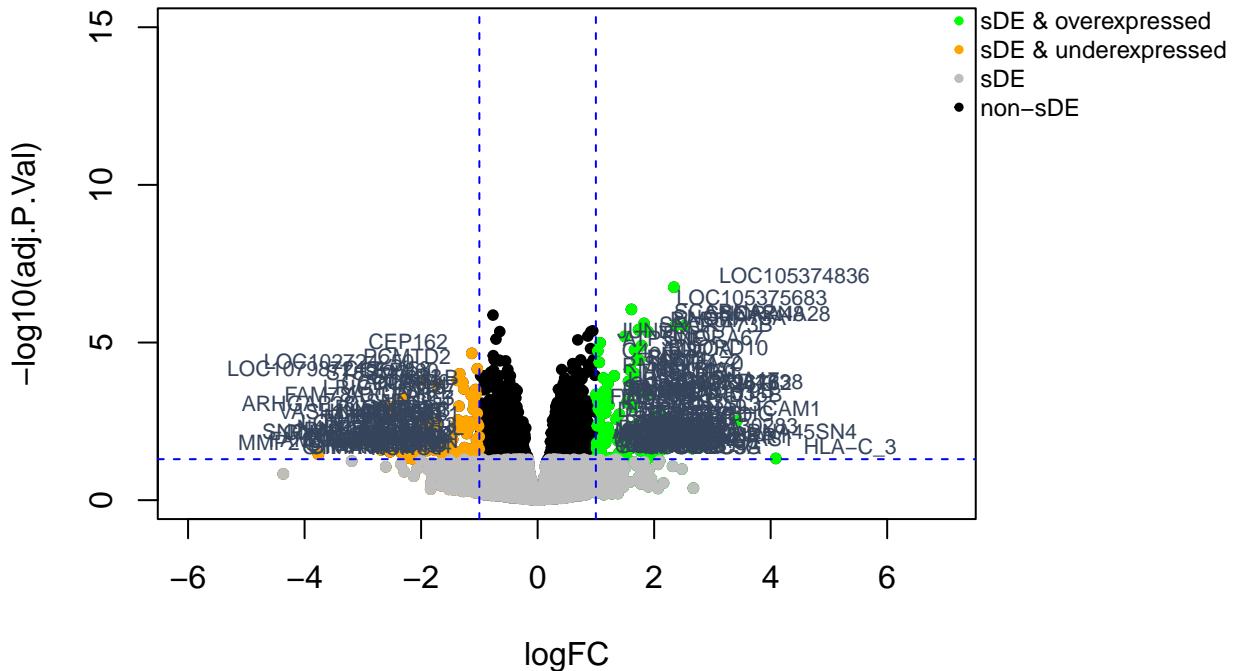
# Volcano plot
par(xpd=T, mar=par()$mar+c(0,0,0,5))
with(tt, plot(logFC, -log10(adj.P.Val), pch = 20, main = "", xlim=c(-6,7), ylim=c(0,15)))
```

```

with(subset(tt, logFC > 1.0), points(logFC, -log10(adj.P.Val), pch = 20, col="green"))
with(subset(tt, logFC < -1.0), points(logFC, -log10(adj.P.Val), pch = 20, col="orange"))
with(subset(tt, -log10(adj.P.Val) < 1.3), points(logFC, -log10(adj.P.Val), pch = 20, col="grey"))

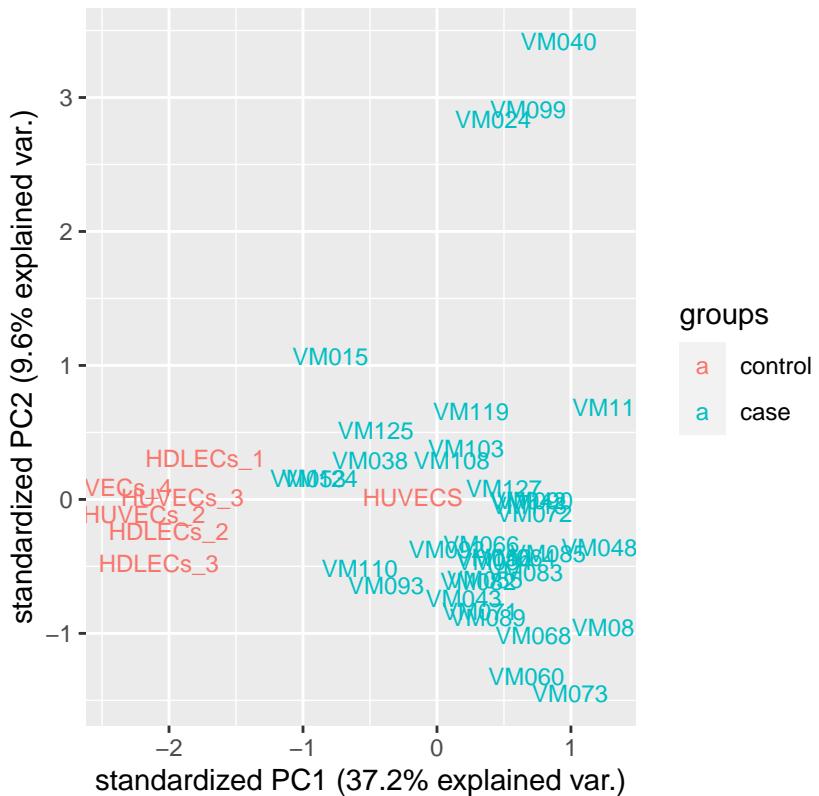
legend("bottomright", cex = .75, inset = c(-0.32,0.75), xjust = 2, yjust = 1,pch = 20,c("sDE & overexpr
with(subset(tt, -log10(adj.P.Val)>1.3 & abs(logFC)>1), textxy(logFC, -log10(adj.P.Val), labs = symbol,
par(xpd=F)
abline(h= 1.3, col = "blue", lty= 2, lwd = 1)
abline(v= c(-1,1), col = "blue", lty= 2, lwd = 1)

```



```
pca_norm_post <- prcomp(as.data.frame(t(assays(se.filt)$logCPM.norm[genes,])) , center=TRUE, scale=TRUE)
ggbiplots(pca_norm_post, var.axes=FALSE, groups=se.filt$Case.Control, labels = colnames(se.filt)) + ggtint
```

PCA based on normalized data and colored by type

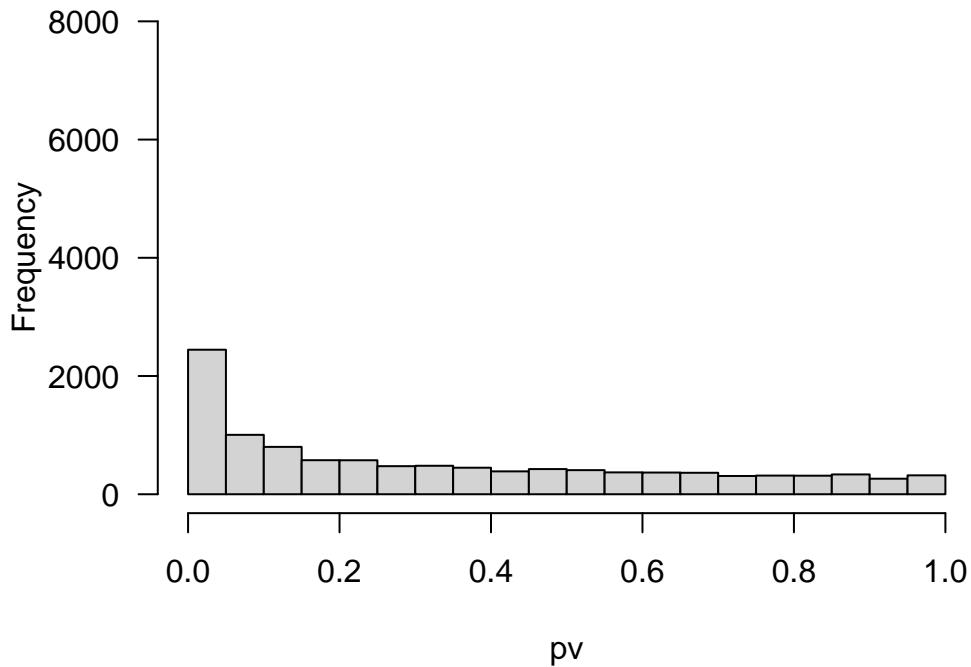


```
## Venous controls vs venous patients
patients_v <- se.filt[, se.filt$Summary.clinic=="venous malformation"]
controls_v <- se.filt[, colnames(assay(se.filt))%in%c("HUVECS", "HUVECs_2", "HUVECs_3", "HUVECs_4")]
se.v <- cbind(patients_v, controls_v)

se.v$Case.Control <- relevel(factor(se.v$Case.Control), ref="control")
mod <- model.matrix(~ se.v$Case.Control, colData(se.v))
mod0 <- model.matrix(~ 1, colData(se.v))
pv <- f.pvalue(assays(se.v)$logCPM.norm, mod, mod0)
sum(p.adjust(pv, method="fdr") < 0.05)

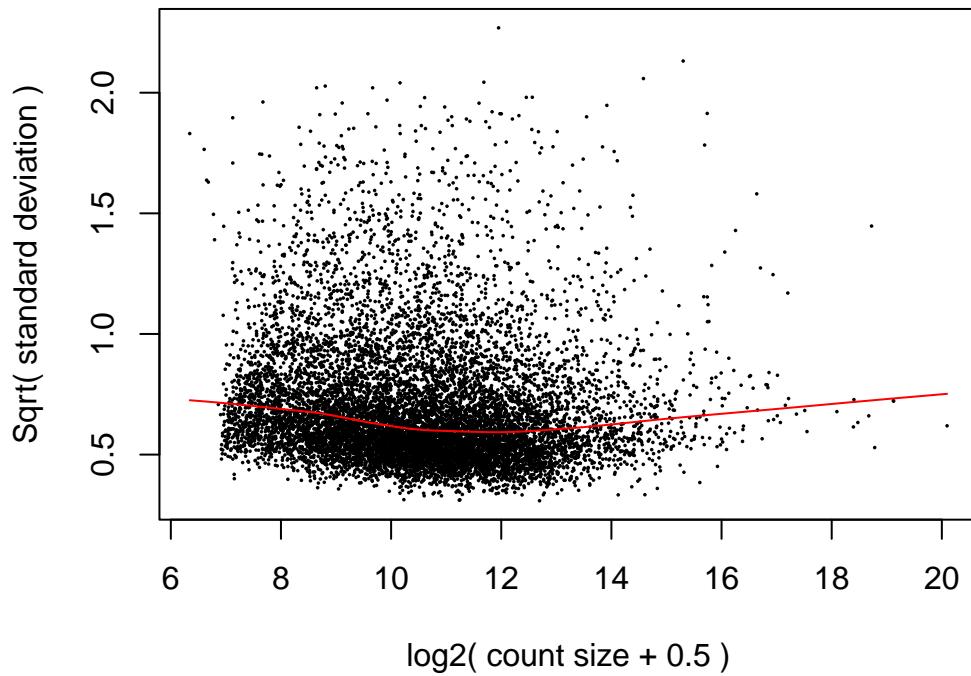
## [1] 516
par(mfrow=c(1, 1))
hist(pv, main="Distribution of expected p-values", las=1, ylim = c(0, 8000))
```

Distribution of expected p-values



```
FDRcutoff <- 0.05  
v <- voom(dge.filt.norm[, colnames(se.v)], mod, plot = TRUE) # Voom is applied to the normalized and fil
```

voom: Mean–variance trend



```
fit <- lmFit(v, mod)  
fit <- eBayes(fit)
```

```

res <- decideTests(fit, p.value = FDRcutoff)
summary(res)

##          (Intercept) se.v$Case.Controlcase
## Down              0                  286
## NotSig            94                 10370
## Up               10892                330

genesmd <- data.frame(symbol = rownames(res), stringsAsFactors = FALSE)
fit$genes <- genesmd
tt <- topTable(fit, coef = 2, n = Inf)
head(tt, 15)

##                               symbol      logFC    AveExpr        t     P.Value
## OPTN           OPTN  1.4496723 6.911360 10.853461 2.425113e-10
## PUS7           PUS7 -1.5905280 4.536192 -8.176858 3.854749e-08
## BMS1P23        BMS1P23 1.5249012 2.191032  8.378891 2.547135e-08
## TYMP           TYMP  2.9657236 2.933494  8.099777 4.521385e-08
## NAXE           NAXE -1.0285522 5.958453 -7.644443 1.179046e-07
## FAM78A         FAM78A -4.3582577 3.557586 -7.334596 2.299713e-07
## ARHGAP28       ARHGAP28 -4.3641702 2.645989 -7.322445 2.361377e-07
## RGS5            RGS5 -7.1796146 4.013660 -7.348957 2.228963e-07
## CEP162          CEP162 -1.5527115 2.303644 -7.057749 4.222816e-07
## GYPC            GYPC  4.1963612 6.706092  6.827312 7.056884e-07
## TENM3           TENM3  3.0280521 5.745024  6.790762 7.660542e-07
## SNHG14          SNHG14 -1.5690487 3.569564 -6.758104 8.244671e-07
## LOC112268052   LOC112268052 1.0820695 4.161498  6.550851 1.318469e-06
## KIAA1614        KIAA1614  1.2047049 3.181407  6.430280 1.736839e-06
## STK10           STK10  0.8988024 6.238566  6.283205 2.436689e-06
##                               adj.P.Val      B
## OPTN           2.664229e-06 13.361890
## PUS7           1.241798e-04  8.774572
## BMS1P23        1.241798e-04  8.620534
## TYMP           1.241798e-04  8.132164
## NAXE           2.590600e-04  7.740234
## FAM78A         3.242761e-04  7.074960
## ARHGAP28       3.242761e-04  7.057426
## RGS5            3.242761e-04  6.966590
## CEP162          5.154651e-04 6.448800
## GYPC            7.547997e-04 6.000918
## TENM3           7.547997e-04 5.928171
## SNHG14          7.547997e-04 5.906309
## LOC112268052   1.114208e-03 5.423076
## KIAA1614        1.362923e-03 5.095149
## STK10           1.729617e-03 4.903638

tt_sign <- tt[tt$adj.P.Val <= 0.05,]
head(tt_sign); nrow(tt_sign)

##                               symbol      logFC    AveExpr        t     P.Value     adj.P.Val
## OPTN           OPTN  1.449672 6.911360 10.853461 2.425113e-10 2.664229e-06
## PUS7           PUS7 -1.590528 4.536192 -8.176858 3.854749e-08 1.241798e-04
## BMS1P23        BMS1P23 1.524901 2.191032  8.378891 2.547135e-08 1.241798e-04
## TYMP           TYMP  2.965724 2.933494  8.099777 4.521385e-08 1.241798e-04
## NAXE           NAXE -1.028552 5.958453 -7.644443 1.179046e-07 2.590600e-04

```

```

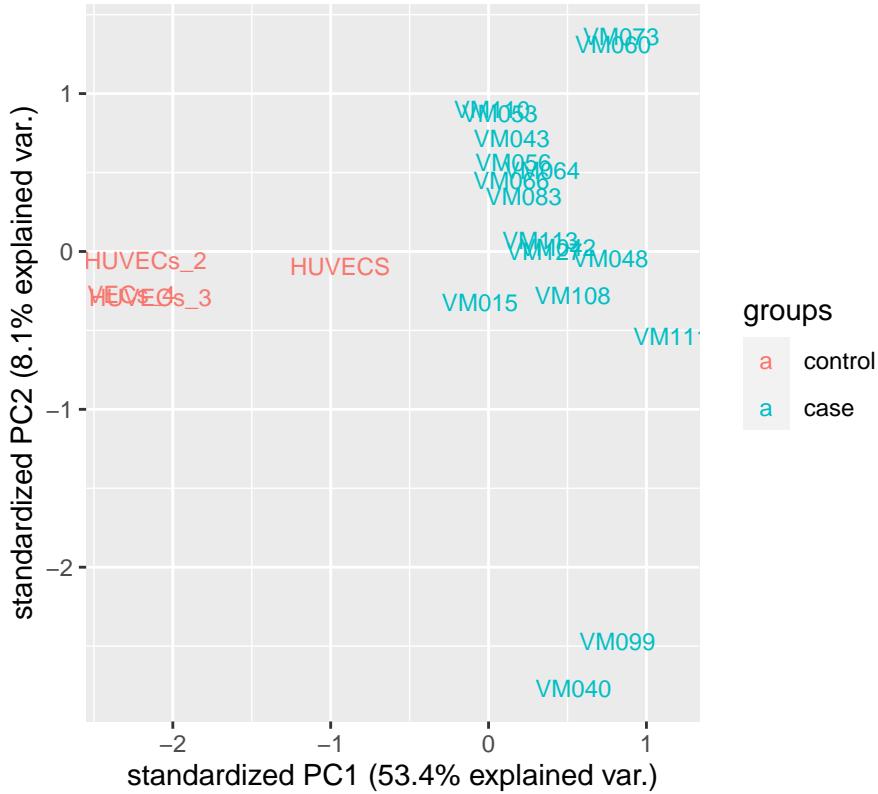
## FAM78A    FAM78A -4.358258 3.557586 -7.334596 2.299713e-07 3.242761e-04
##                      B
## OPTN      13.361890
## PUS7      8.774572
## BMS1P23   8.620534
## TYMP      8.132164
## NAXE      7.740234
## FAM78A    7.074960
## [1] 616
sum(tt$adj.P.Val < 0.05)

## [1] 616
genes_v <- (tt$symbol[(tt$adj.P.Val < 0.05)])
genes_v_or <- genes[order(genes)]

#Venous PCA
pca_norm_v <- prcomp(as.data.frame(t(assays(se.v)$logCPM.norm[genes_v,])), center=TRUE, scale=TRUE)
ggbioplot(pca_norm_v, var.axes=FALSE, groups=se.v$Case.Control, labels = colnames(se.v)) + ggtitle("PCA")

```

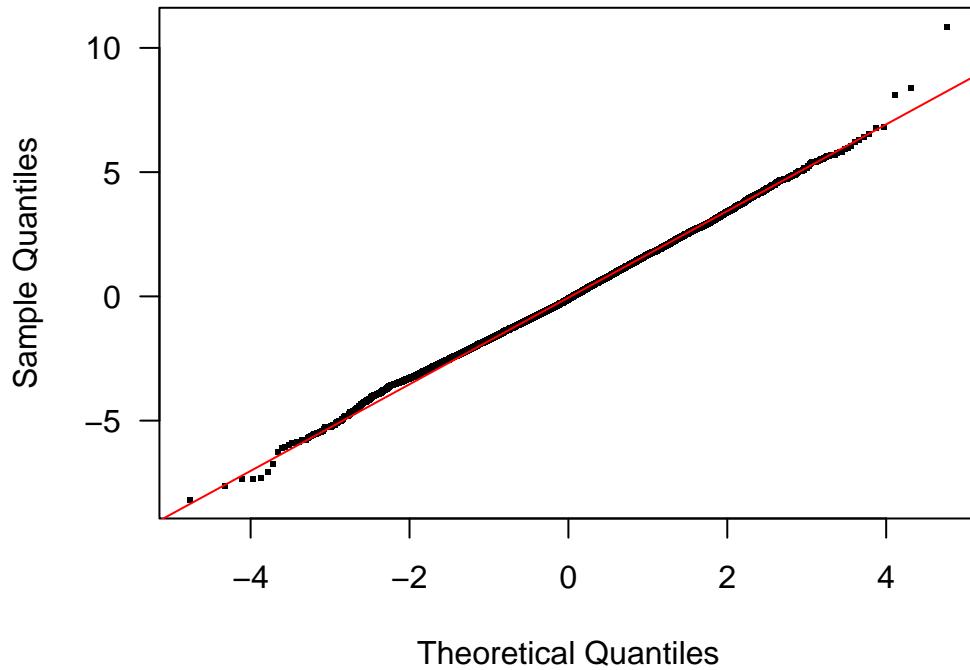
PCA based on venous normalized data and colored by type



```

# Venous Q-Q plot
{qqt(fit$t[, 2], df = fit$df.prior + fit$df.residual, main = "", pch = ".", cex = 3, las = 1)
 qqline(fit$t[, 2], col = "red")}

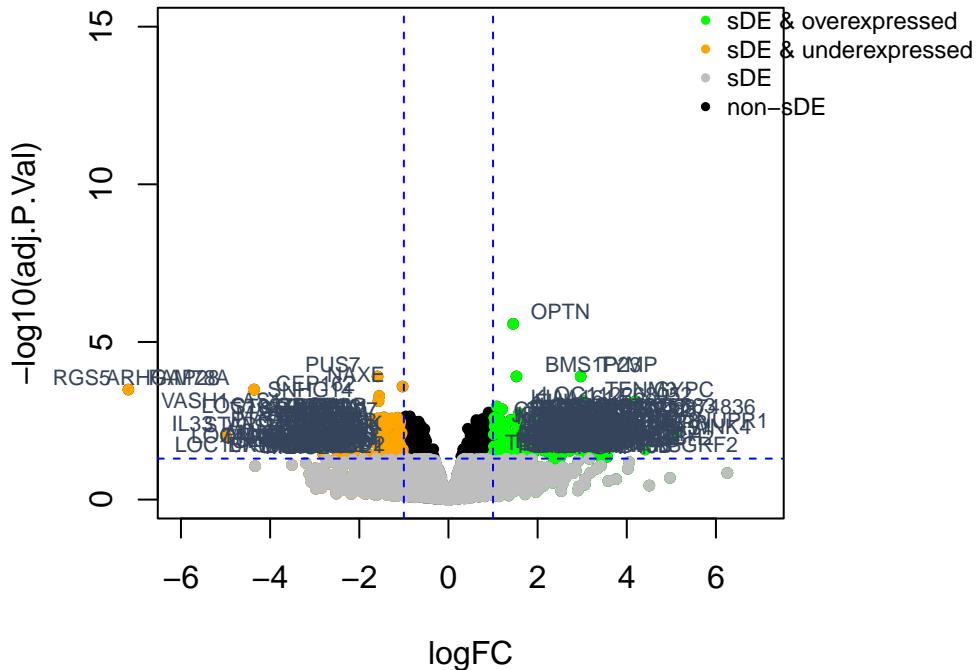
```



```
# Venous volcano plot
par(xpd=T, mar=par()$mar+c(0,0,0,5))
with(tt, plot(logFC, -log10(adj.P.Val), pch = 20, main = "", xlim=c(-6,7), ylim=c(0,15)))

with(subset(tt, logFC > 1.0), points(logFC, -log10(adj.P.Val), pch = 20, col="green"))
with(subset(tt, logFC < -1.0), points(logFC, -log10(adj.P.Val), pch = 20, col="orange"))
with(subset(tt, -log10(adj.P.Val) < 1.3), points(logFC, -log10(adj.P.Val), pch = 20, col="grey"))

legend("bottomright", cex = .75, inset = c(-0.32,0.75), xjust = 2, yjust = 1,pch = 20,c("sDE & overexpr"))
with(subset(tt, -log10(adj.P.Val)>1.3 & abs(logFC)>1), textxy(logFC, -log10(adj.P.Val), labs = symbol, cex = 1.5))
par(xpd=F)
abline(h= 1.3, col = "blue", lty= 2, lwd = 1)
abline(v= c(-1,1), col = "blue", lty= 2, lwd = 1)
```



```

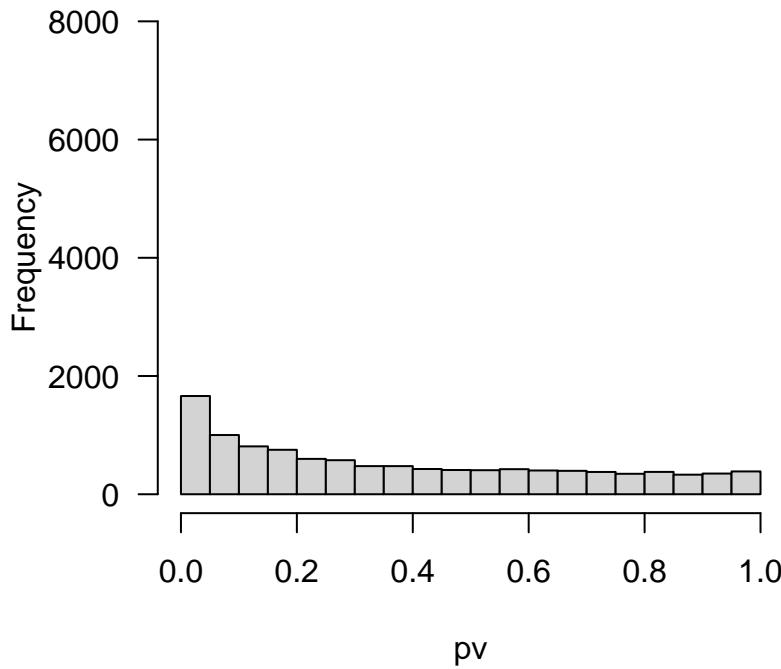
## Lymphatic controls vs lymphatic patients
patients_1 <- se.filt[se.filt$Summary.clinic=="lymphatic malformation"]
controls_1 <- se.filt[,colnames(assay(se.filt))%in%c("HDLECs_1", "HDLECs_2", "HDLECs_3")]
se.1 <- cbind(patients_1, controls_1)

se.1$Case.Control <- relevel(factor(se.1$Case.Control), ref="control")
mod <- model.matrix(~ se.1$Case.Control, colData(se.1))
mod0 <- model.matrix(~ 1, colData(se.1))
pv <- f.pvalue(assays(se.1)$logCPM.norm, mod, mod0)
sum(p.adjust(pv, method="fdr") < 0.05)

## [1] 3
par(mfrow=c(1, 1))
hist(pv, main="Distribution of expected p-values", las=1, ylim = c(0, 8000))

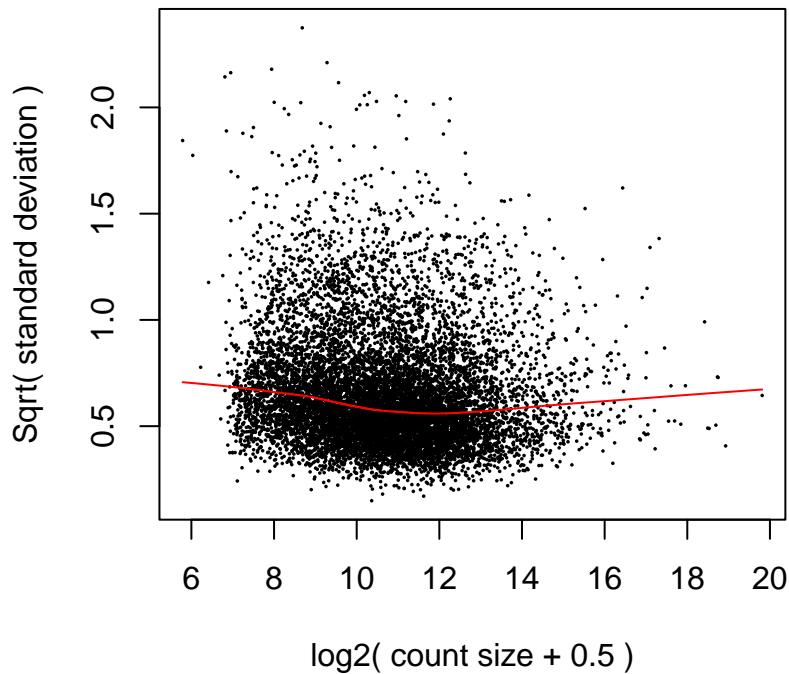
```

Distribution of expected p-values



```
FDRcutoff <- 0.05  
v <- voom(dge.filt.norm[, colnames(se.1)], mod, plot = TRUE) # Voom is applied to the normalized and fil
```

voom: Mean–variance trend



```
fit <- lmFit(v, mod)  
fit <- eBayes(fit)
```

```

res <- decideTests(fit, p.value = FDRcutoff)
summary(res)

##          (Intercept) se.l$Case.Controlcase
## Down              2                      1
## NotSig            117                     10983
## Up               10867                   2

genesmd <- data.frame(symbol = rownames(res), stringsAsFactors = FALSE)
fit$genes <- genesmd
tt <- topTable(fit, coef = 2, n = Inf)
head(tt, 15)

##           symbol      logFC    AveExpr       t     P.Value
## LOC105374836 LOC105374836 2.1269217 5.115864 11.287160 8.951972e-06
## ALDH1A1        ALDH1A1   -7.1909543 6.571429 -11.202855 9.415297e-06
## VEGFC          VEGFC    4.5052545 3.446348 11.103768 9.995029e-06
## PLAT           PLAT     3.7762490 6.772080  9.478434 2.870813e-05
## GALNT15        GALNT15  -3.8707288 3.809518 -9.285876 3.287009e-05
## MALAT1          MALAT1  1.5635450 11.137811 8.379739 6.432727e-05
## TALAM1          TALAM1  1.5698808 11.119218 8.331322 6.679063e-05
## PLEKHF1         PLEKHF1  2.0081447 2.492529 8.357896 6.542565e-05
## LOC283788       LOC283788 -2.4242466 3.611502 -7.929517 9.189502e-05
## YTHDC2          YTHDC2  -1.5222626 5.793277 -7.640949 1.165266e-04
## SCARNA17        SCARNA17 2.0328384 1.744736 8.400894 6.328326e-05
## SGIP1            SGIP1   -3.4491502 3.459497 -7.229569 1.655878e-04
## F2RL1            F2RL1  -3.6797407 4.231972 -7.137010 1.796024e-04
## CHKA             CHKA   -0.9457737 4.088061 -6.957098 2.108306e-04
## SPRY1            SPRY1  -1.8189603 5.024944 -6.789898 2.454139e-04
##           adj.P.Val       B
## LOC105374836  0.03660180 3.581247
## ALDH1A1        0.03660180 3.210552
## VEGFC          0.03660180 3.025642
## PLAT           0.07222217 2.785205
## GALNT15        0.07222217 2.513183
## MALAT1          0.08152910 1.951187
## TALAM1          0.08152910 1.925263
## PLEKHF1         0.08152910 1.762459
## LOC283788       0.10095587 1.758634
## YTHDC2          0.11637825 1.705399
## SCARNA17        0.08152910 1.683733
## SGIP1            0.12631393 1.273541
## F2RL1            0.12631393 1.237505
## CHKA             0.12631393 1.130177
## SPRY1            0.12631393 1.037408

tt_sign <- tt[tt$adj.P.Val <= 0.05,]
head(tt_sign); nrow(tt_sign)

##           symbol      logFC    AveExpr       t     P.Value adj.P.Val
## LOC105374836 LOC105374836 2.126922 5.115864 11.28716 8.951972e-06 0.0366018
## ALDH1A1        ALDH1A1   -7.190954 6.571429 -11.20286 9.415297e-06 0.0366018
## VEGFC          VEGFC    4.505255 3.446348 11.10377 9.995029e-06 0.0366018
##           B
## LOC105374836  3.581247

```

```

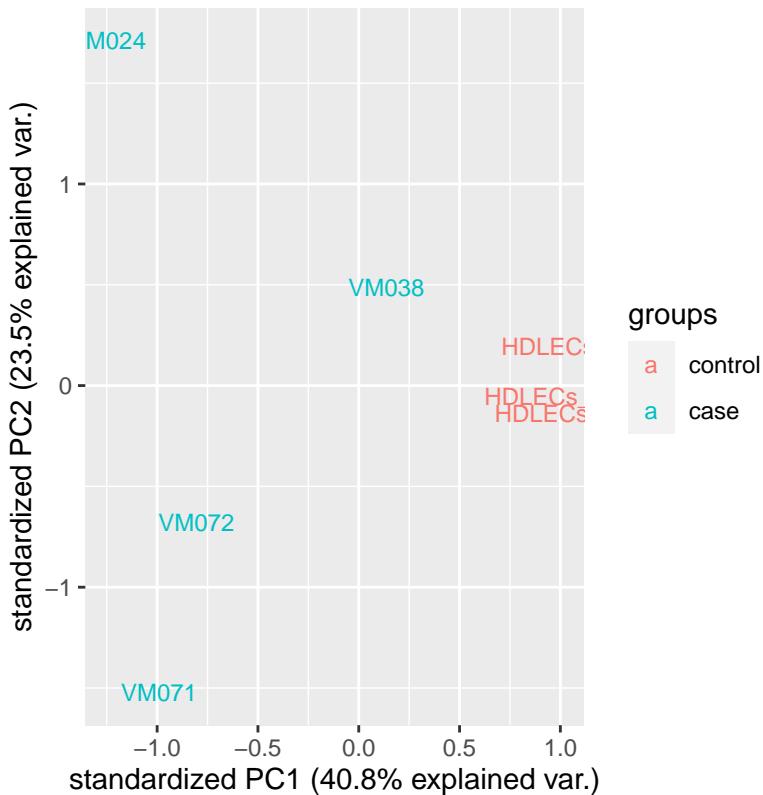
## ALDH1A1      3.210552
## VEGFC       3.025642
## [1] 3
sum(tt$adj.P.Val < 0.05)

## [1] 3
genes_l <- (tt$symbol[(tt$adj.P.Val < 0.05)])
genes_l_or <- genes[order(genes)]

#Lymphatic PCA
pca_norm_l <- prcomp(as.data.frame(t(assays(se.1)$logCPM.norm[genes_v,])), center=TRUE, scale=TRUE)
ggbiplots(pca_norm_l, var.axes=FALSE, groups=se.1$Case.Control, labels = colnames(se.1)) + ggtitle("PCA")

```

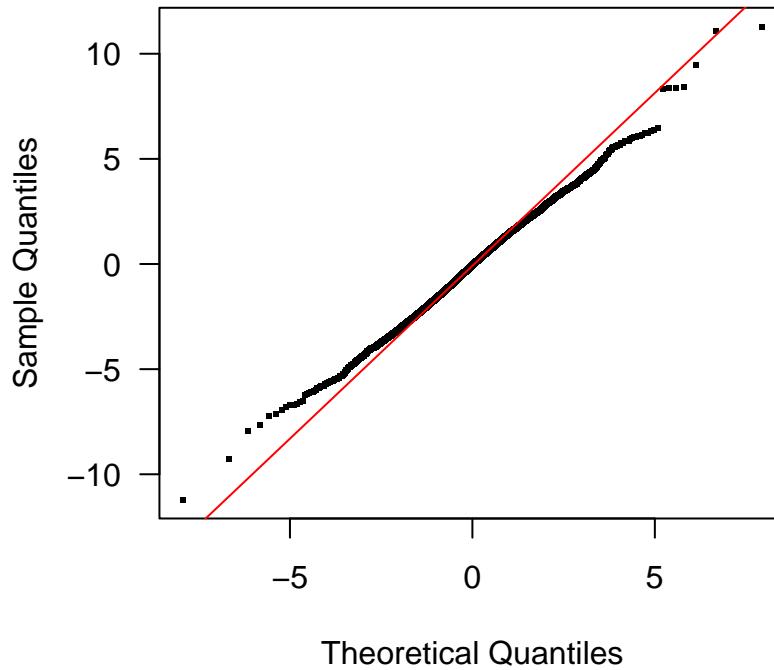
PCA based on venous normalized data and colored by type



```

# Lymphatic Q-Q plot
{qqt(fit$t[, 2], df = fit$df.prior + fit$df.residual, main = "", pch = ".", cex = 3, las = 1)
 qqline(fit$t[, 2], col = "red")}

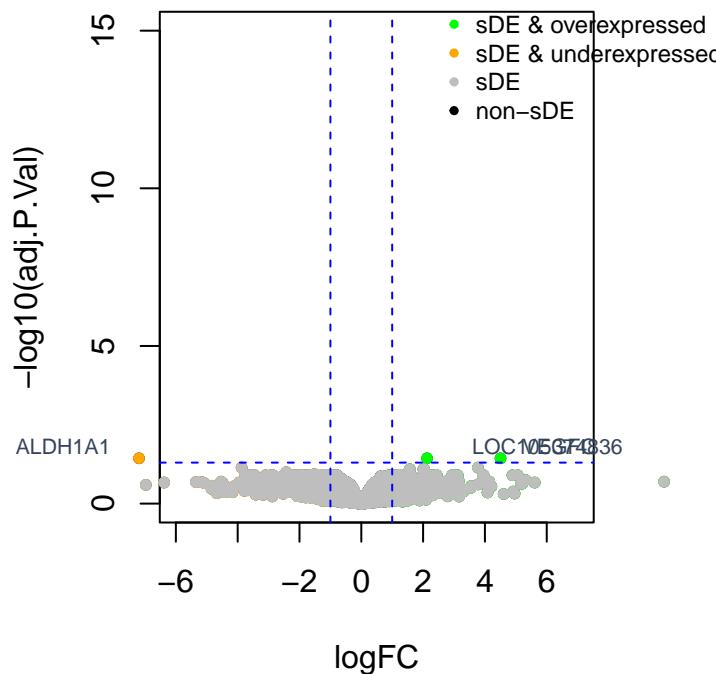
```



```
# Lymphatic volcano plot
par(xpd=T, mar=par()$mar+c(0,0,0,5))
with(tt, plot(logFC, -log10(adj.P.Val), pch = 20, main = "", xlim=c(-6,7), ylim=c(0,15)))

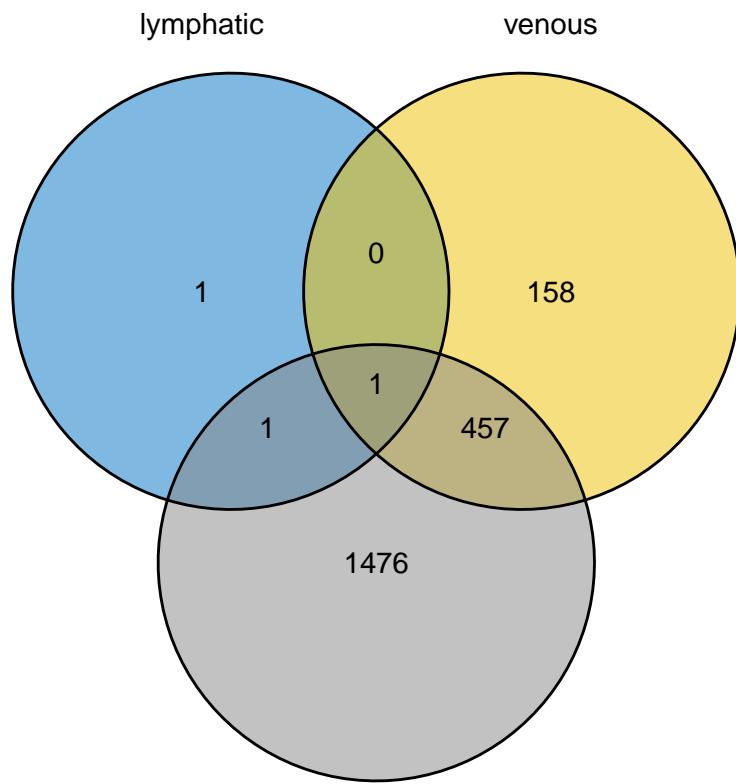
with(subset(tt, logFC > 1.0), points(logFC, -log10(adj.P.Val), pch = 20, col="green"))
with(subset(tt, logFC < -1.0), points(logFC, -log10(adj.P.Val), pch = 20, col="orange"))
with(subset(tt, -log10(adj.P.Val) < 1.3), points(logFC, -log10(adj.P.Val), pch = 20, col="grey"))

legend("bottomright", cex = .75, inset = c(-0.32,0.75), xjust = 2, yjust = 1,pch = 20,c("sDE & overexpr"))
with(subset(tt, -log10(adj.P.Val)>1.3 & abs(logFC)>1), textxy(logFC, -log10(adj.P.Val), labs = symbol, cex = 1.5))
par(xpd=F)
abline(h= 1.3, col = "blue", lty= 2, lwd = 1)
abline(v= c(-1,1), col = "blue", lty= 2, lwd = 1)
```



```
#Compare DE genes from the three groups
genes_lst <- list(genes_l, genes_v, genes)
names(genes_lst) <- c("lymphatic", "venous", "all patients vs all controls")
saveRDS(genes_lst, file "~/Vascular_Disease/MG-04_Illumina_totalRNASeq/preprocess/DEgenes")

ggvenn(
  genes_lst, columns=names(genes_lst),
  fill_color = c("#0073C2FF", "#EFC000FF", "#868686FF", "#CD534CFF"),
  stroke_size = 0.5, set_name_size = 4, show_percentage = FALSE
)
```



all patients vs all controls