

SPEAQeasy Differential Expression Analysis

July 10, 2020

Contents

1	Analysis	1
1.1	Load required libraries	1
1.2	Load data and prepare directories to place outputs in	2
1.3	statistics PCs	2
1.4	Stats vs. race and brain region	2
1.5	Explore and visualize gene expression	7
1.6	Modeling	29
1.7	Check plots	32
1.8	Gene ontology	36
1.9	Visualize differentially expressed genes	43

1 Analysis

The following analysis explores a `RangedSummarizedExperiment` object from the SPEAQeasy pipeline. Note that we will use a modified version of the object, which resolved sample identity issues which were present in the raw output from SPEAQeasy. This object also includes phenotype data added after resolving identity issues. Though SPEAQeasy produces objects for several feature types (genes, exons, exon-exon junctions), we will demonstrate an example analysis for just genes. We will perform differential expression across some typical variables of interest (e.g. sex, age, race) and show how to perform principal component analysis (PCA) and visualize findings with plots.

1.1 Load required libraries

```
library("SummarizedExperiment")
library("recount")
library("edgeR")
library("limma")
library("jaffelab") # GitHub: LieberInstitute/jaffelab
library("RColorBrewer")
library("clusterProfiler")
library("org.Hs.eg.db")
library("pheatmap")
library("here")
```

1.2 Load data and prepare directories to place outputs in

For those who ran SPEAQeasy from the example FASTQ data set, the `RangedSummarizedExperiment` will have a different path, as specified with the `--output` flag.

```
# Load the RSE gene object
load(here("rse_speaqeasy.RData"), verbose = TRUE)

## Loading objects:
##   rse_gene

# Create directories to organize outputs from this analysis
dir.create(here("DE_analysis", "pdfs"), showWarnings = FALSE)
dir.create(here("DE_analysis", "tables"), showWarnings = FALSE)
dir.create(here("DE_analysis", "rdas"), showWarnings = FALSE)
```

1.3 statistics PCs

Here we are using principal component analysis to control for the listed variables impact on expression. This will be later added into our linear model.

```
col_names = c('trimmed', 'numReads', 'numMapped', 'numUnmapped',
              'overallMapRate', 'concordMapRate', 'totalMapped', 'mitoMapped',
              'mitoRate', 'totalAssignedGene')
statsPca = prcomp(as.data.frame(colData(rse_gene)[,col_names]))
rse_gene$PC = statsPca$x[,1]
getPcaVars(statsPca)[1]
```

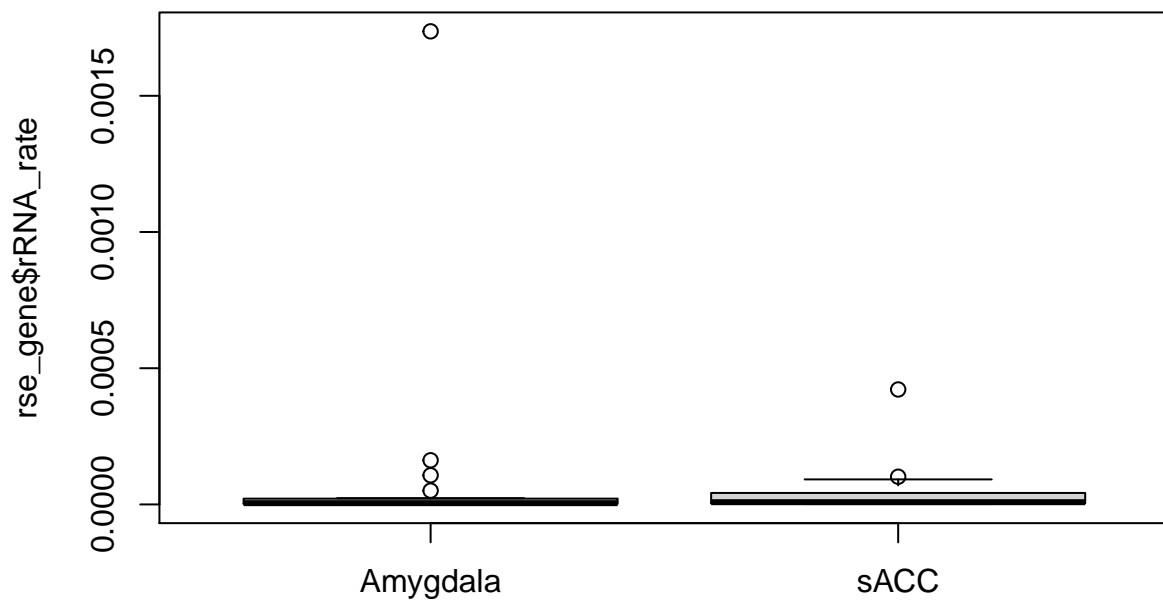
```
## [1] 87.3
```

1.4 Stats vs. race and brain region

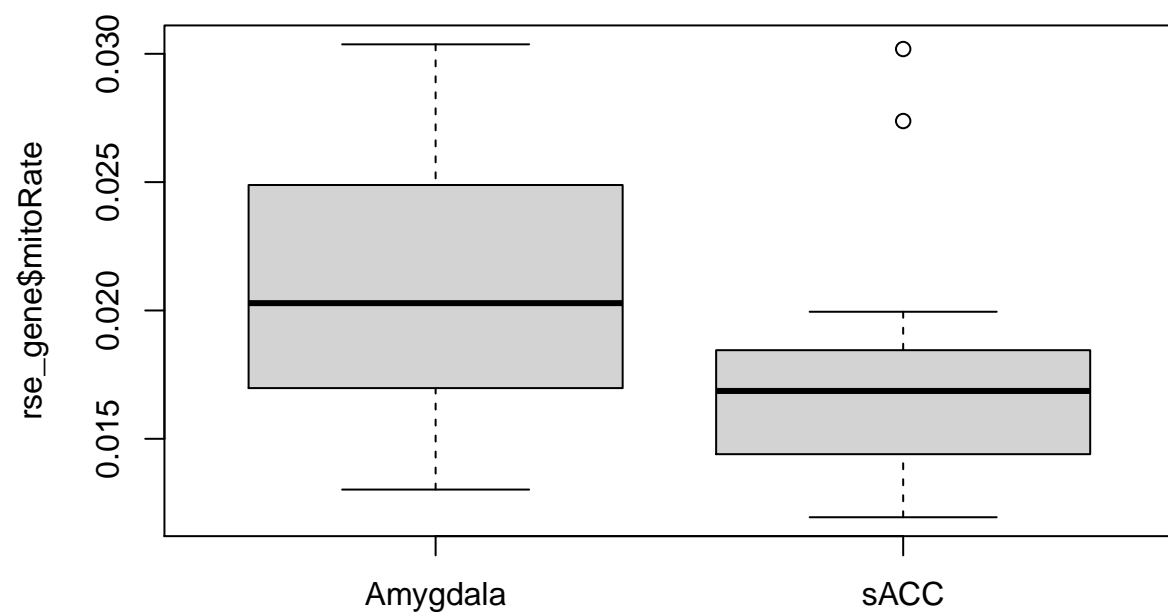
```
# Filter for expressed
rse_gene = rse_gene[rowMeans(getRPKM(rse_gene,"Length")) > 0.2,]

# Check if certain statistics changed by race or region

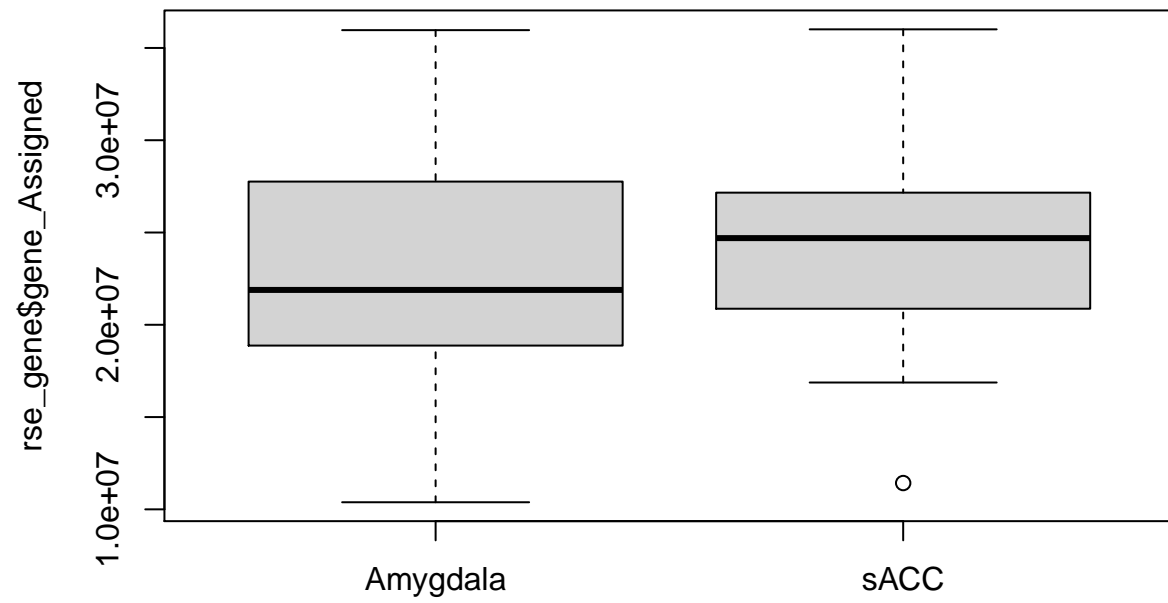
# Display box plots here
boxplot(rse_gene$rRNA_rate ~ rse_gene$BrainRegion,xlab="")
```



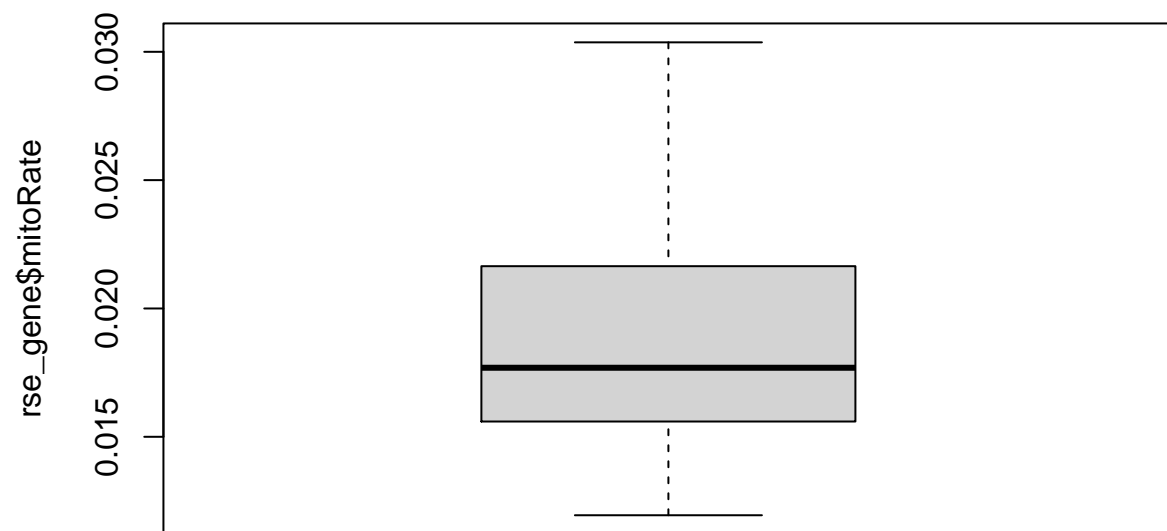
```
boxplot(rse_gene$mitoRate ~ rse_gene$BrainRegion,xlab="")
```



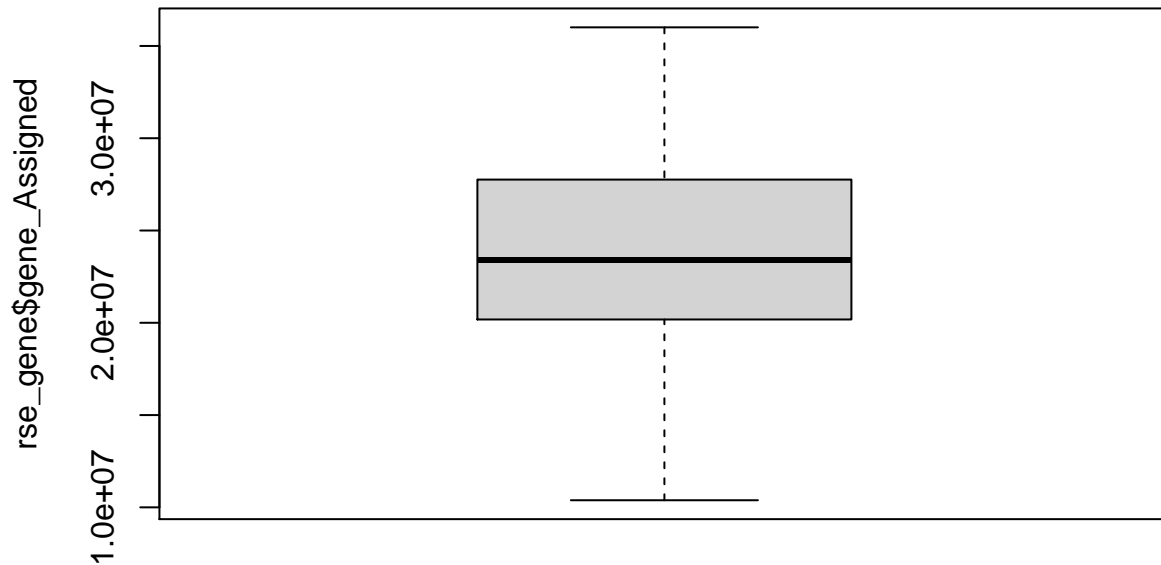
```
boxplot(rse_gene$gene_Assigned ~ rse_gene$BrainRegion,xlab="")
```



```
boxplot(rse_gene$mitoRate ~ rse_gene$Race, las=3, xlab="")
```



```
boxplot(rse_gene$gene_Assigned ~ rse_gene$Race, las=3, xlab="")
```



```
# Save box plots to PDF
pdf(file = here("DE_analysis", "pdfs", "Region_Race_cellcheck.pdf"))
boxplot(rse_gene$rRNA_rate ~ rse_gene$BrainRegion,xlab="")
boxplot(rse_gene$mitoRate ~ rse_gene$BrainRegion,xlab="")
boxplot(rse_gene$gene_Assigned ~ rse_gene$BrainRegion,xlab="")
boxplot(rse_gene$mitoRate ~ rse_gene$Race,las=3,xlab="")
boxplot(rse_gene$gene_Assigned ~ rse_gene$Race,las=3,xlab="")
dev.off()
```

```
## pdf
## 2
```

1.5 Explore and visualize gene expression

```
# Explore gene expression
geneExprs = log2(getRPKM(rse_gene,"Length")+1)
pca = prcomp(t(geneExprs))
pca_vars = getPcaVars(pca)
pca_vars_lab = paste0("PC", seq(along=pca_vars), ": ",
  pca_vars, "% Var Expl")

# Group together code for generating plots of interest
generate_plots = function() {
```

```

par(mar=c(8,6,2,2),cex.axis=1.8,cex.lab=1.8)
palette(brewer.pal(4,"Dark2"))

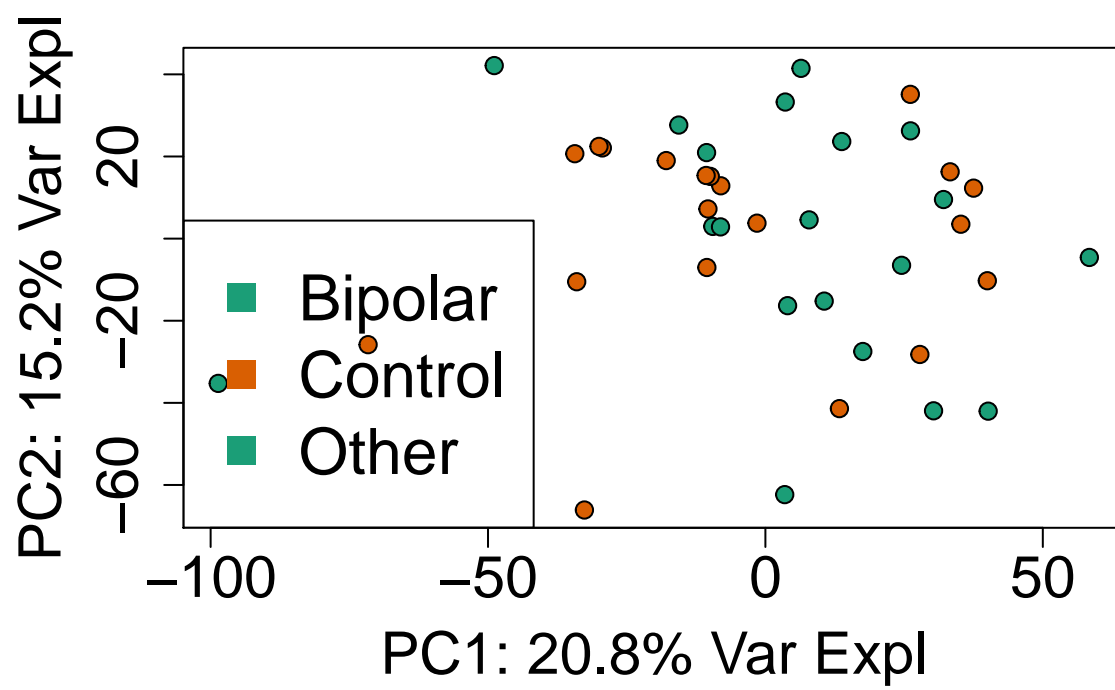
# PC1 vs. PC2
plot(pca$x, pch=21, bg=factor(rse_gene$PrimaryDx),cex=1.2,
     xlab = pca_vars_lab[1], ylab = pca_vars_lab[2])
legend("bottomleft", levels(rse_gene$PrimaryDx), col=1:2, pch=15,cex=2)

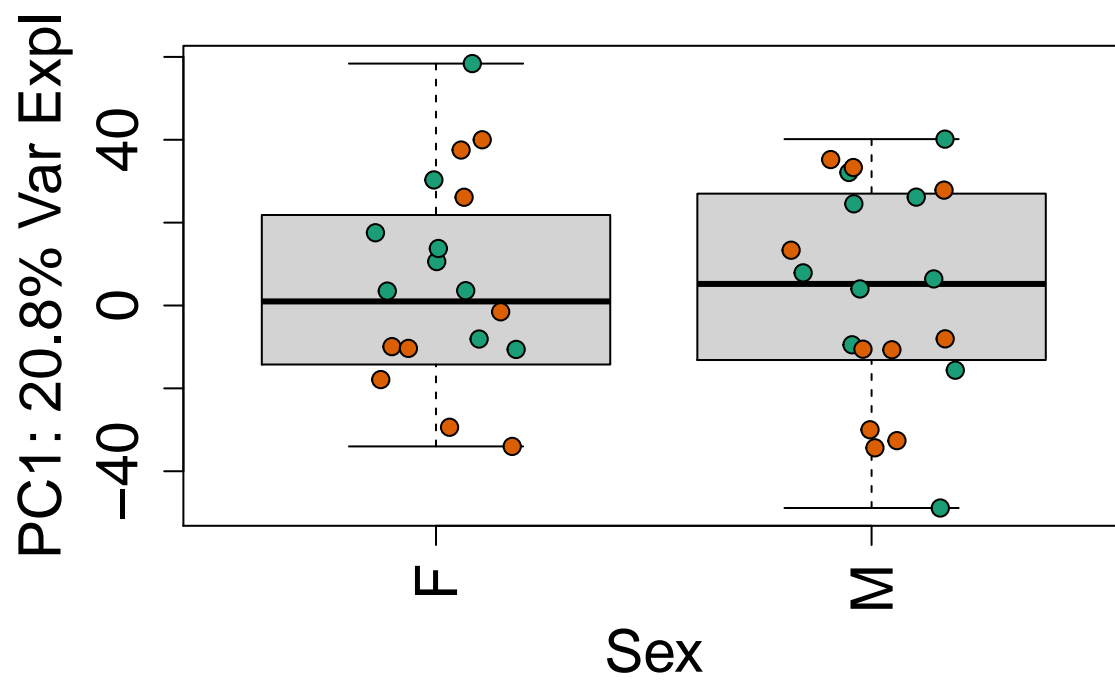
# By line
for(i in 1:10) {
  boxplot(pca$x[,i] ~ rse_gene$Sex,
          ylab=pca_vars_lab[i], las = 3,xlab="Sex",outline=FALSE)
  points( pca$x[,i] ~ jitter(as.numeric(factor(rse_gene$Sex))),
          pch = 21, bg = rse_gene$PrimaryDx,cex=1.2)
}

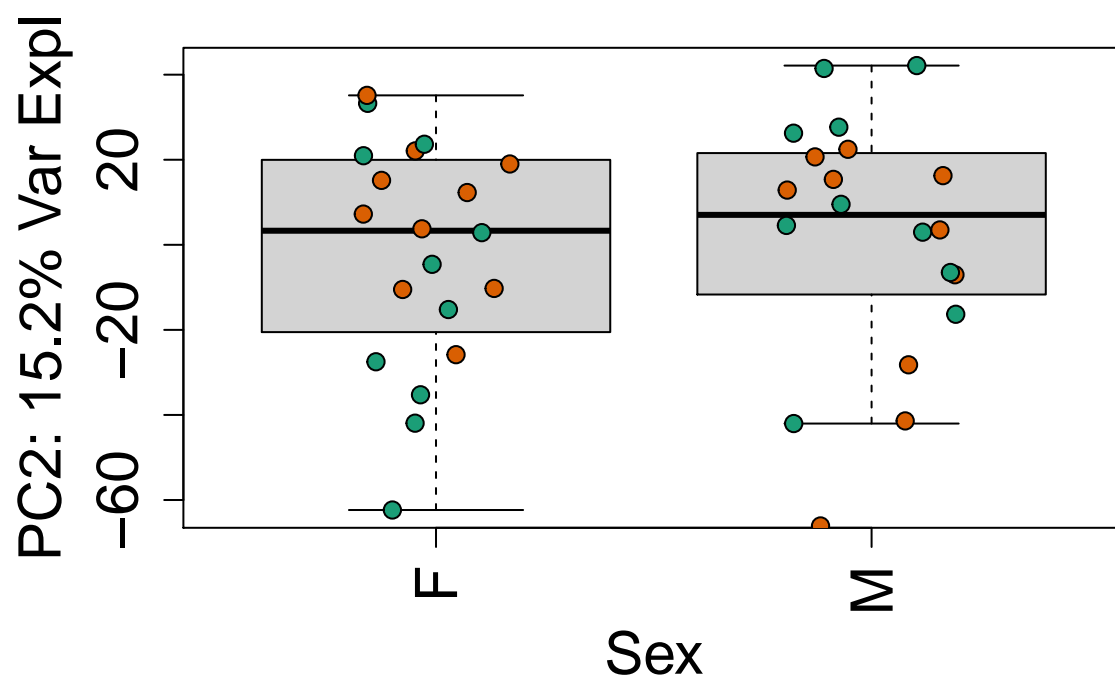
# By experiment
for(i in 1:10) {
  boxplot(pca$x[,i] ~ rse_gene$Race,
          ylab=pca_vars_lab[i], las = 3,xlab="Race",outline=FALSE)
  points( pca$x[,i] ~ jitter(as.numeric(factor(rse_gene$Race))),
          pch = 21, bg = rse_gene$PrimaryDx,cex=1.2)
}
}

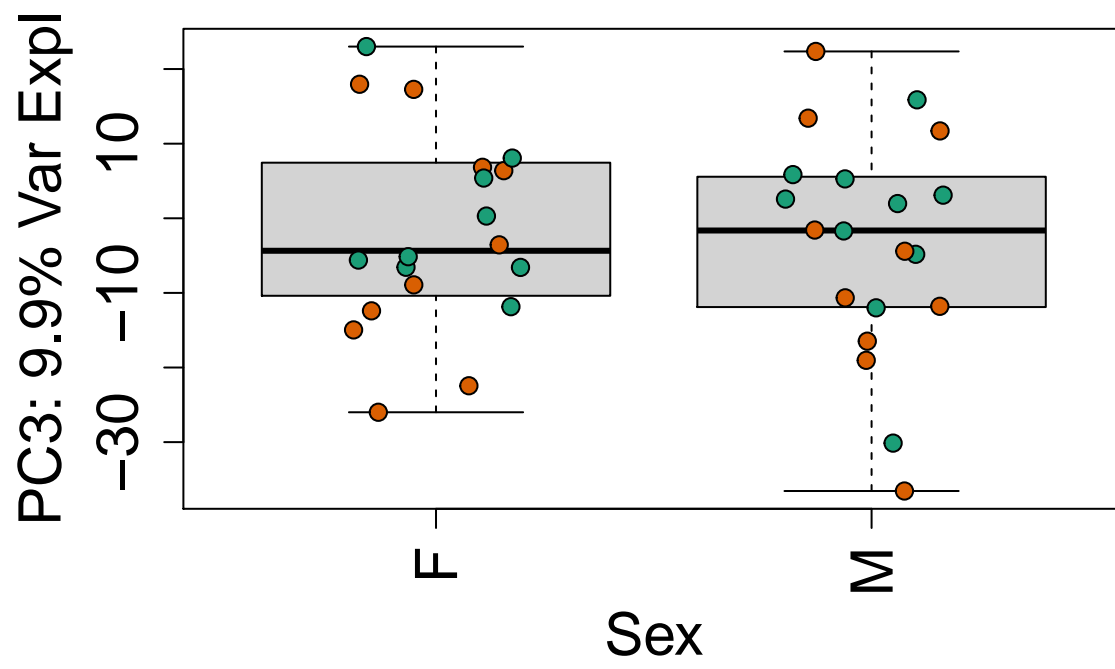
# Display plots
generate_plots()

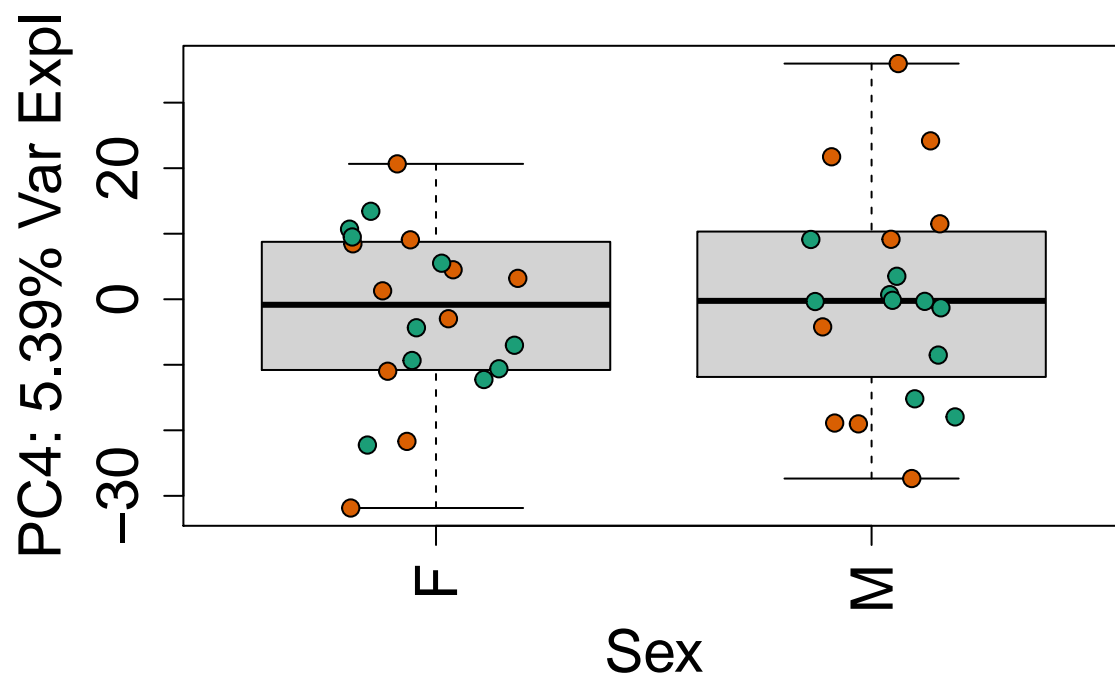
```

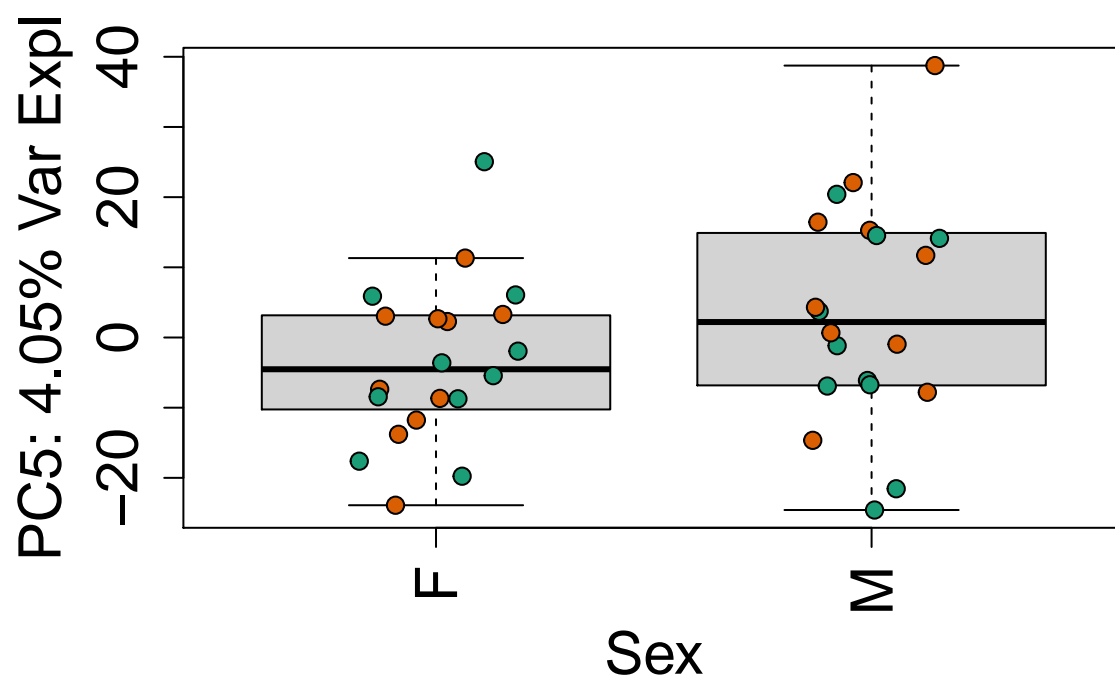



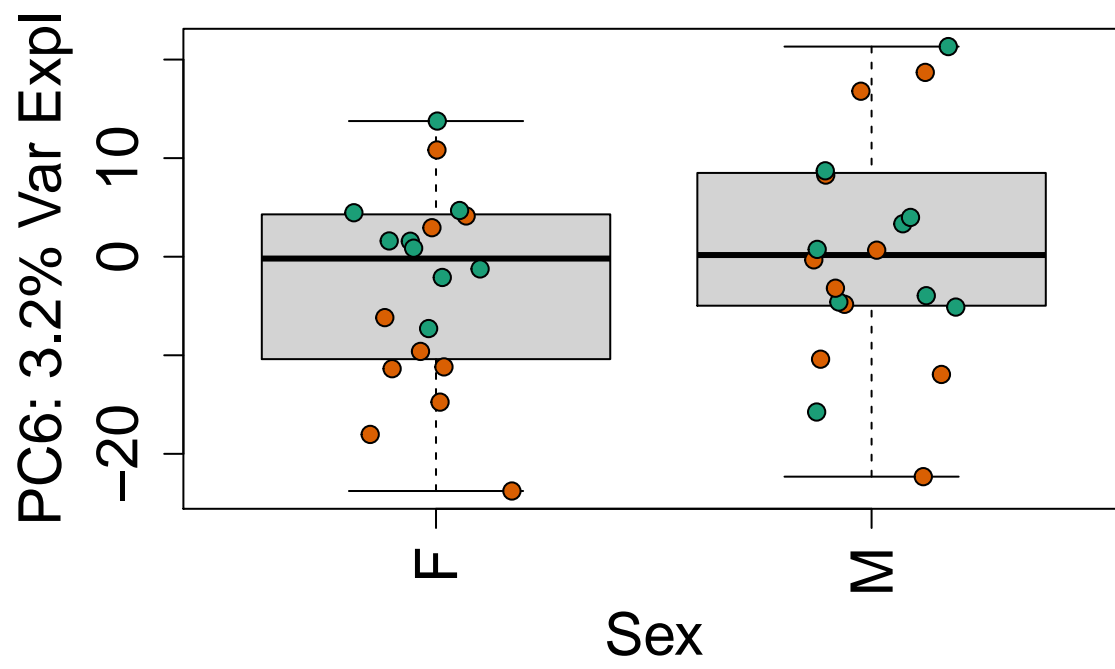


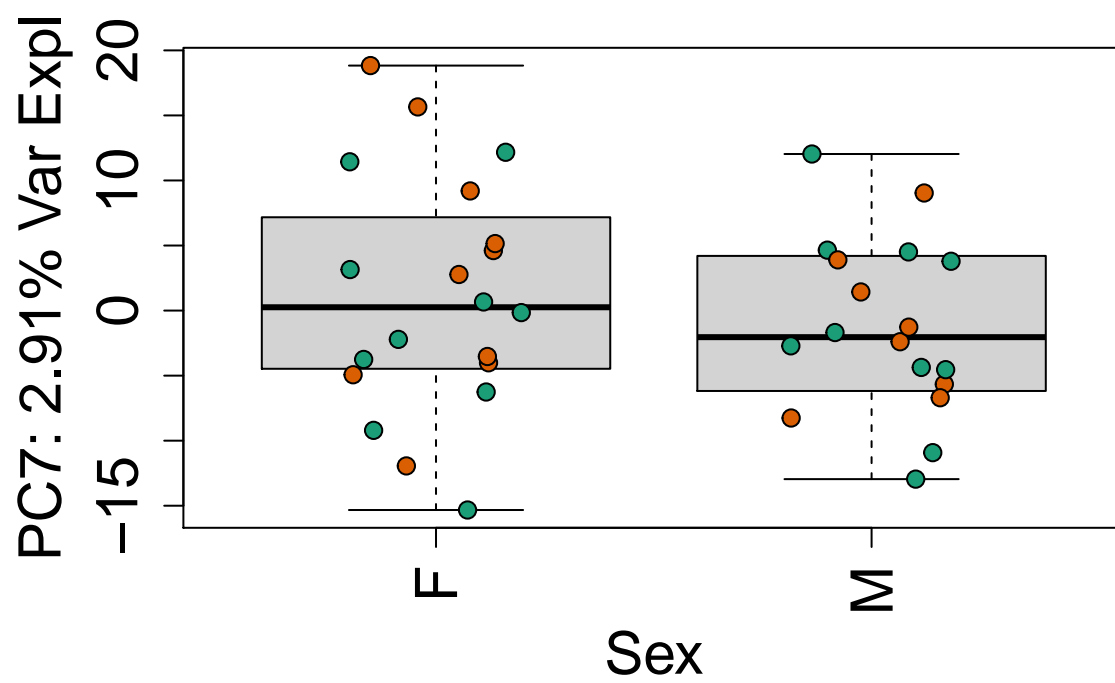


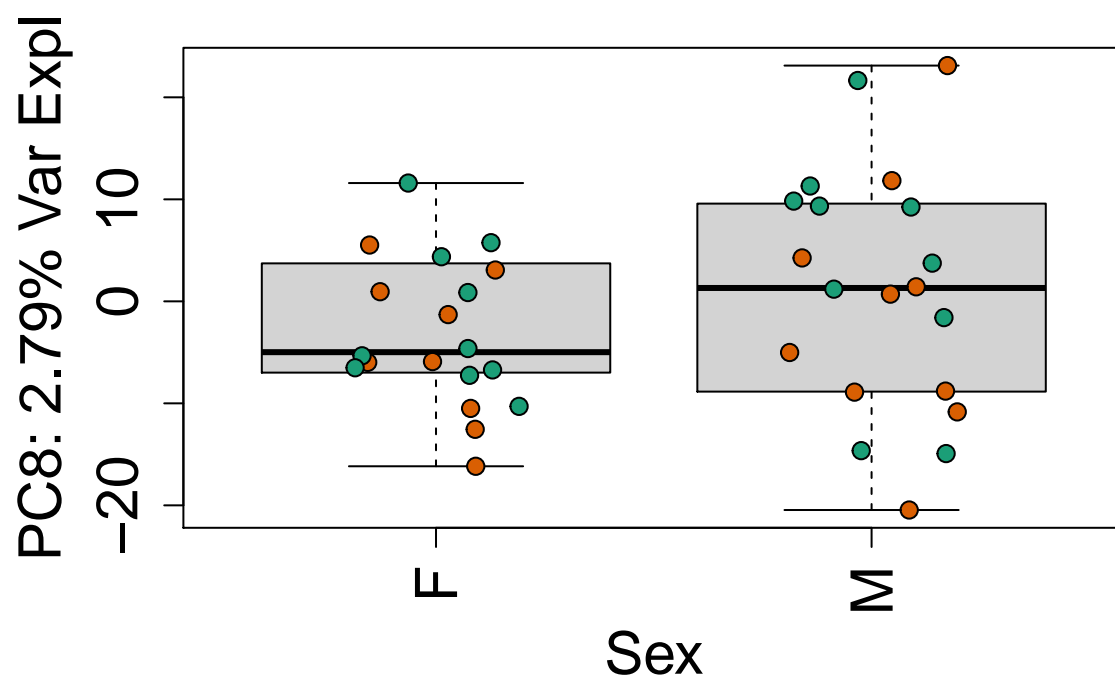


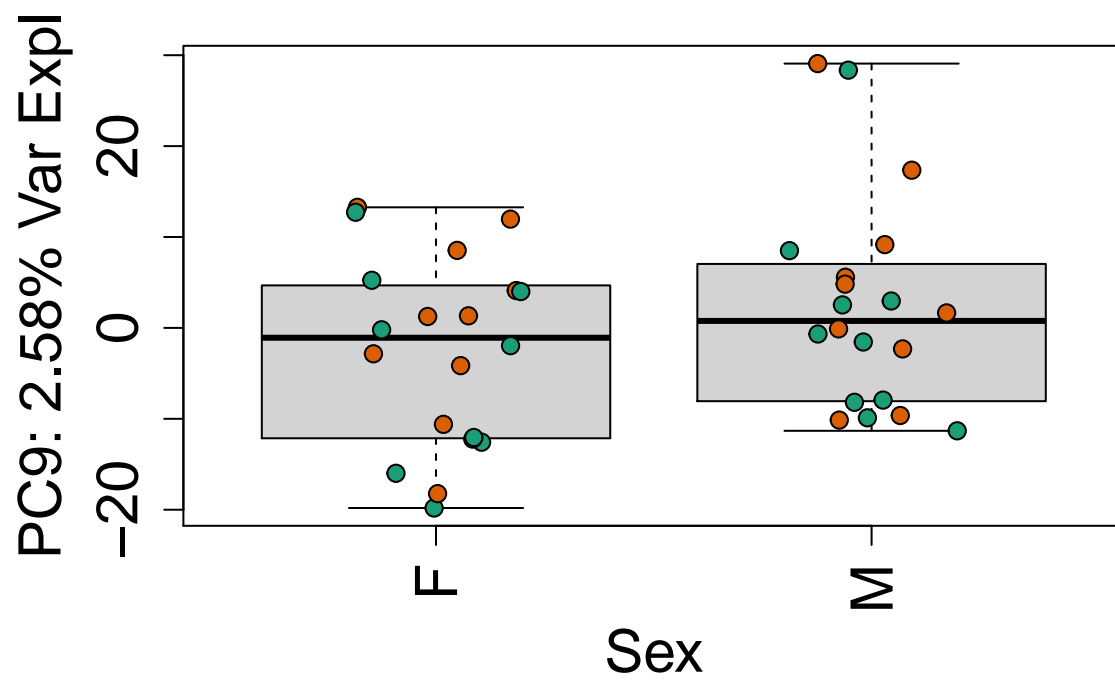


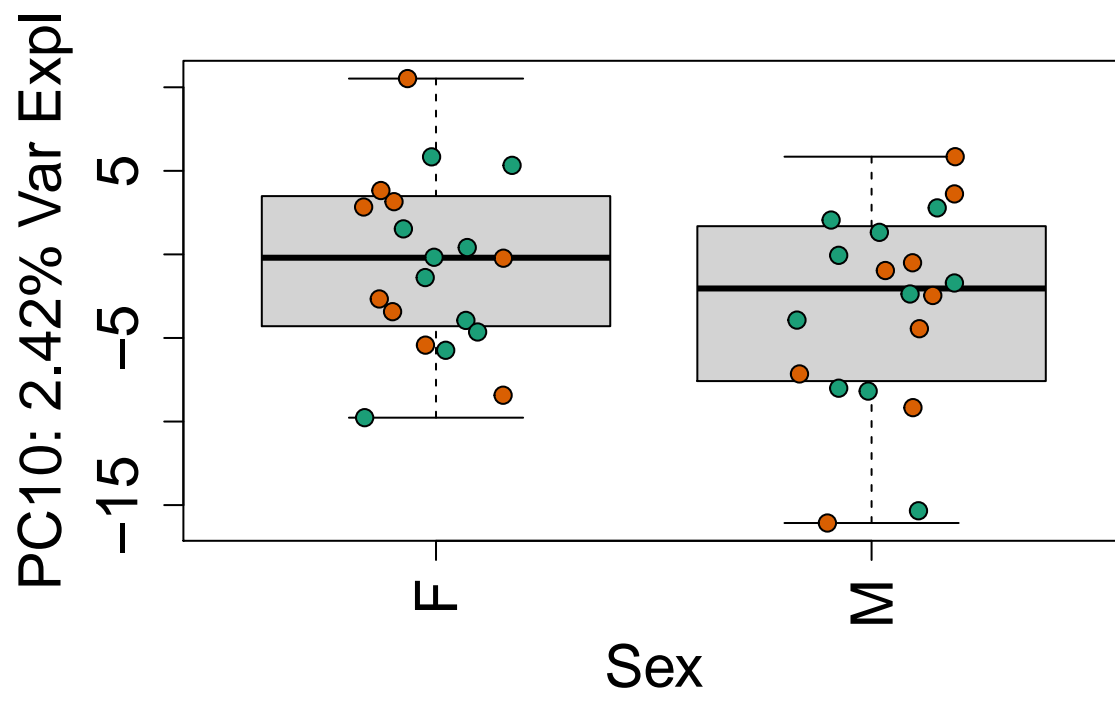


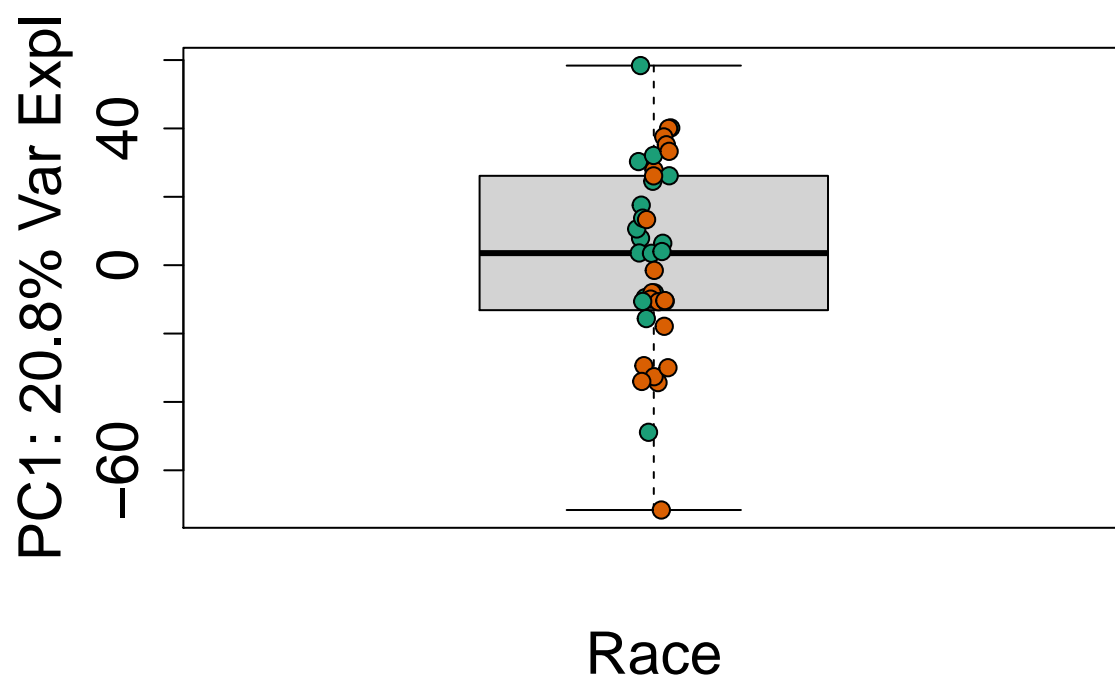


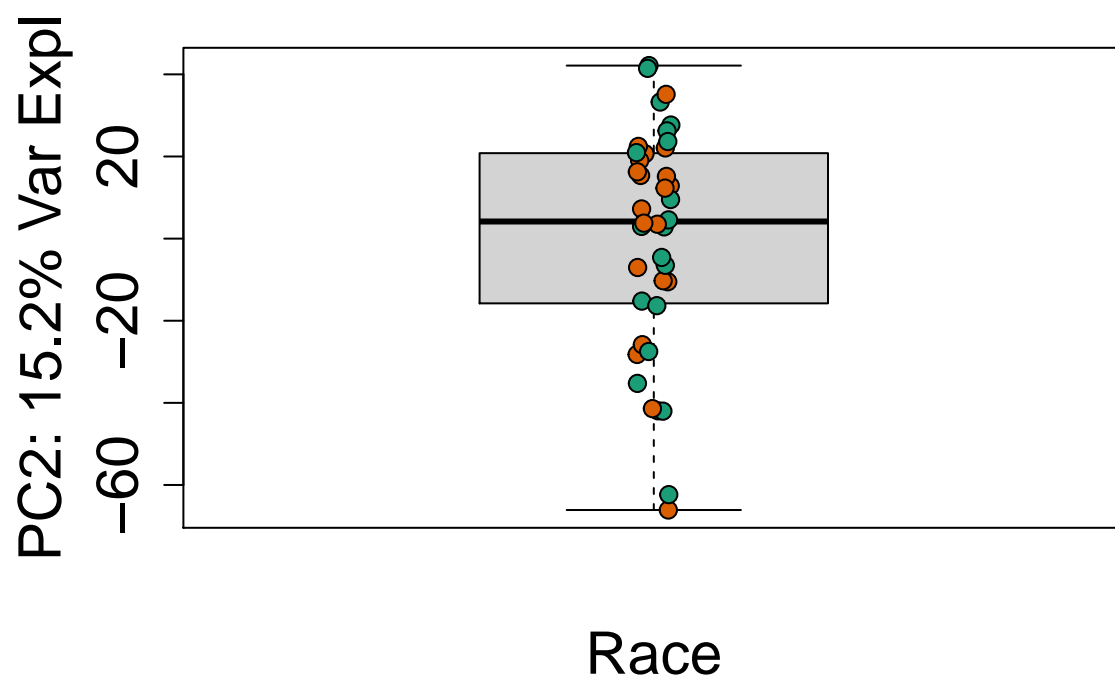


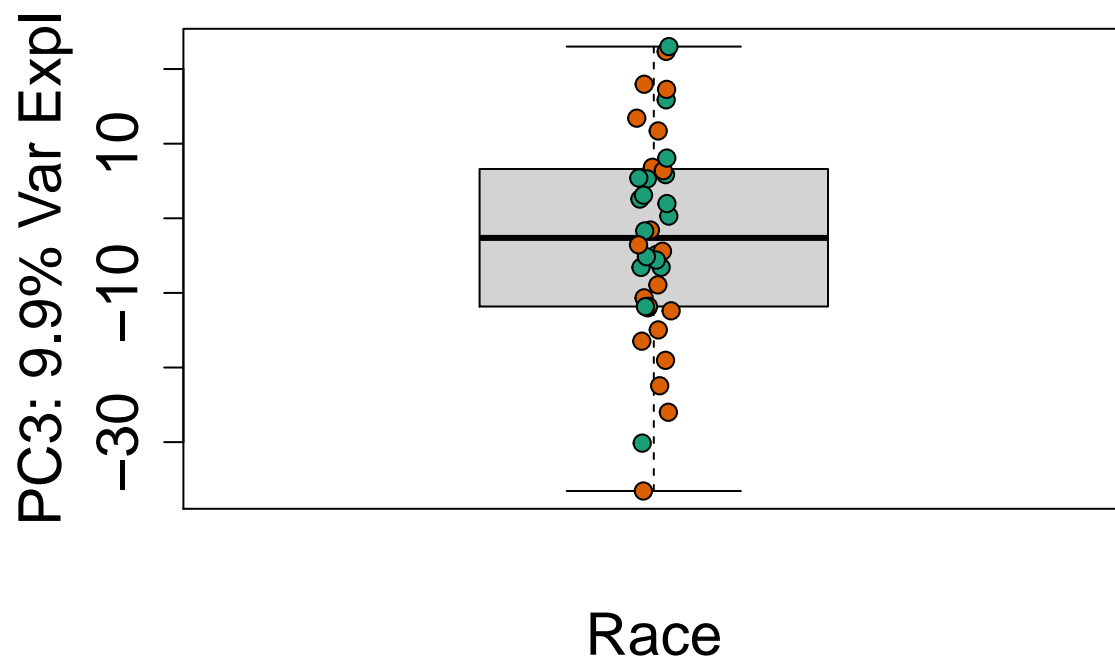


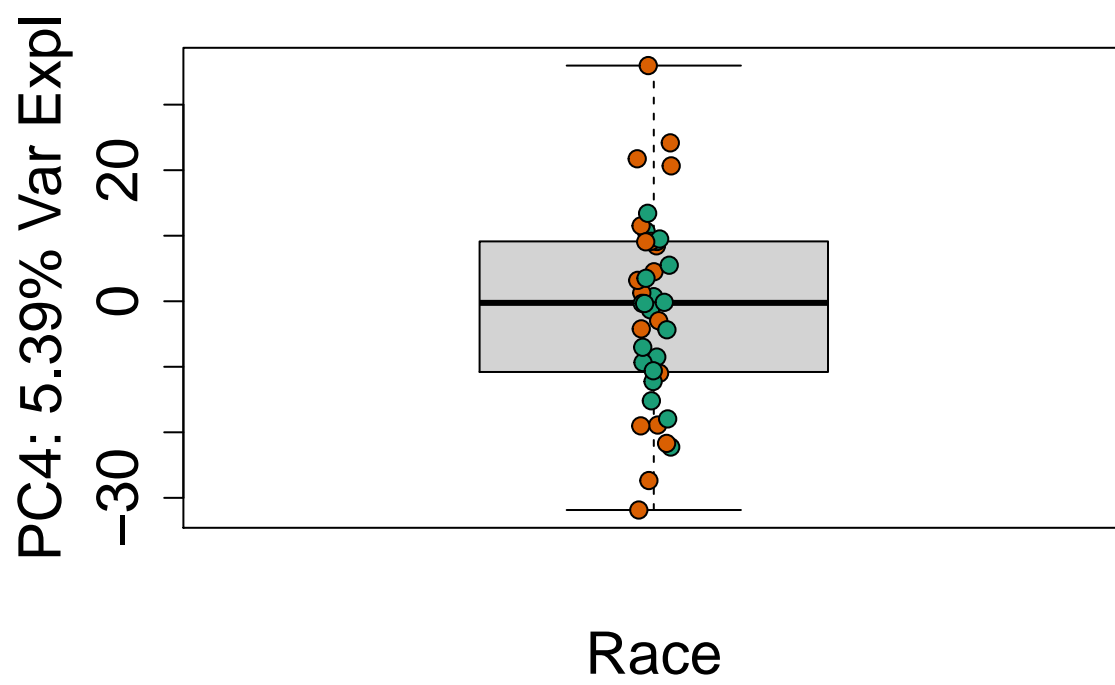


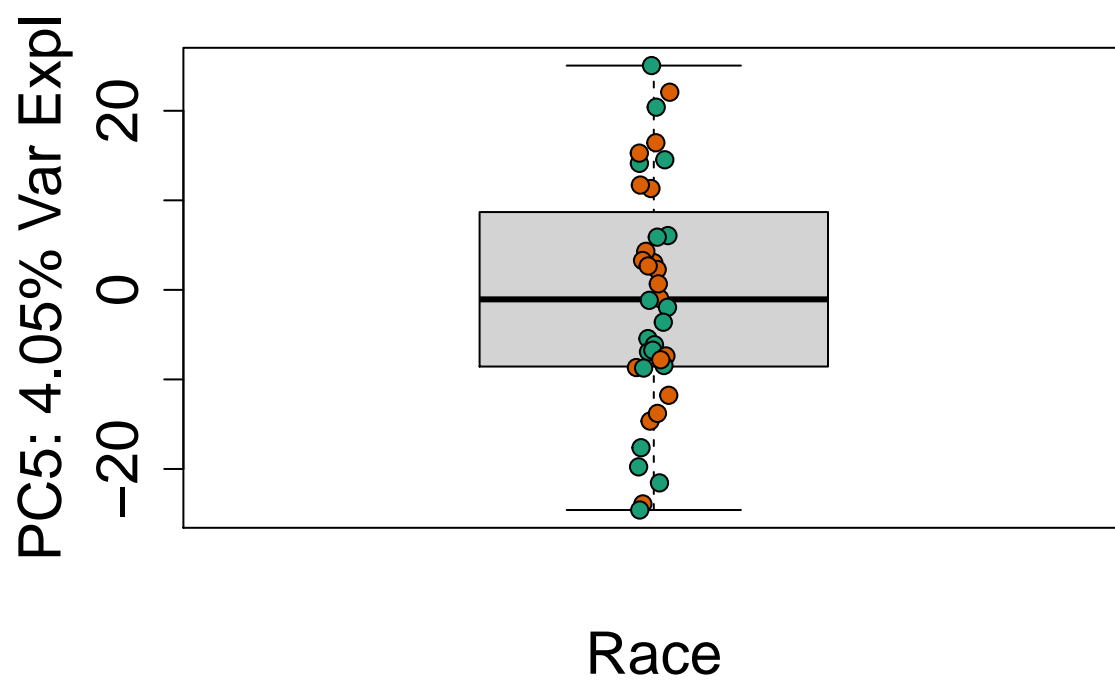


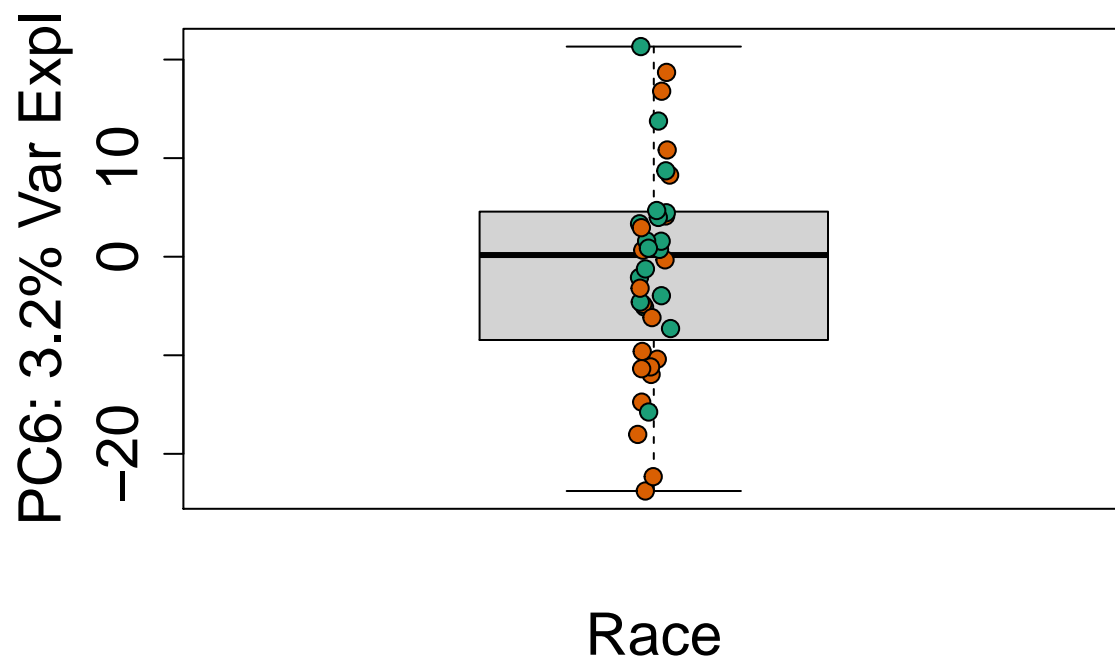


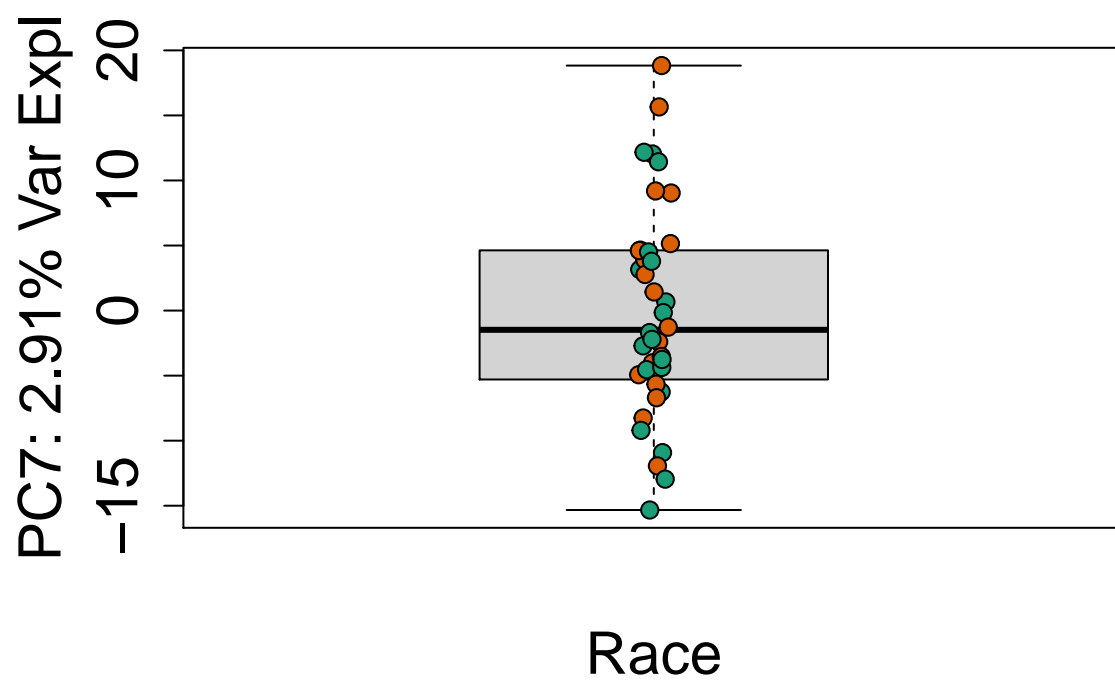


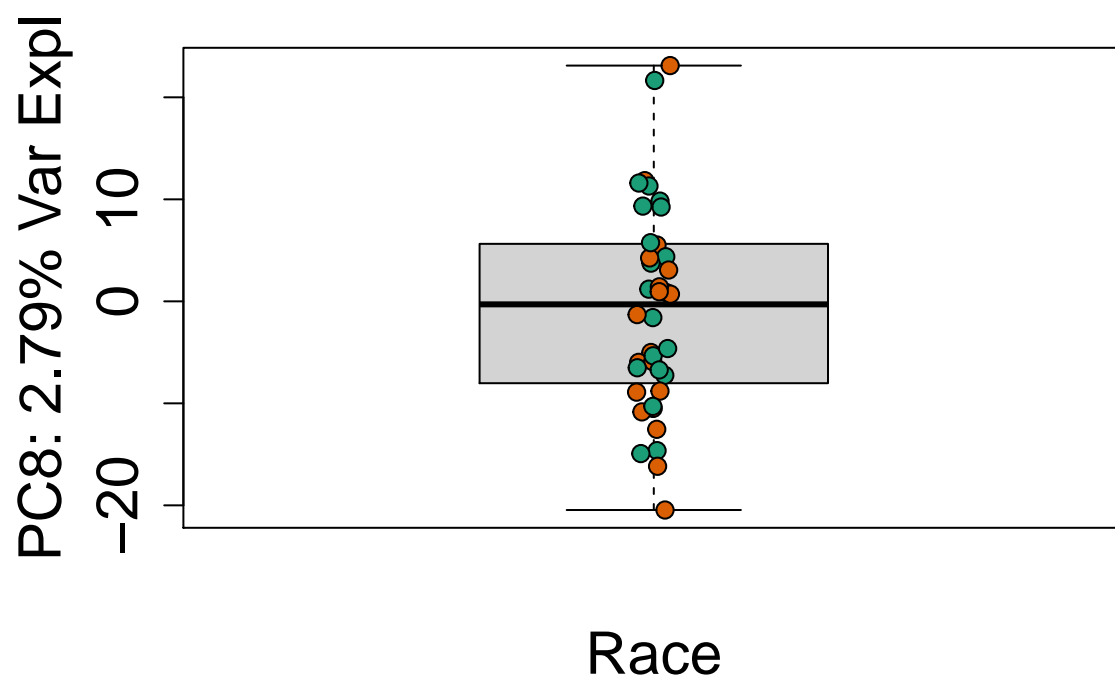


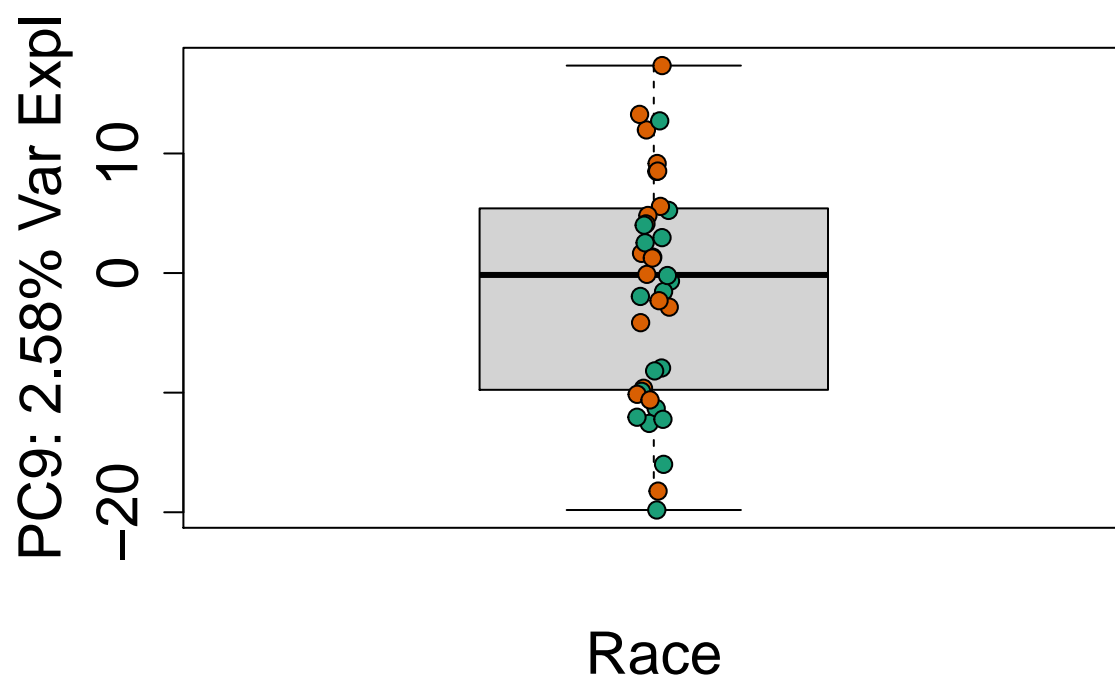


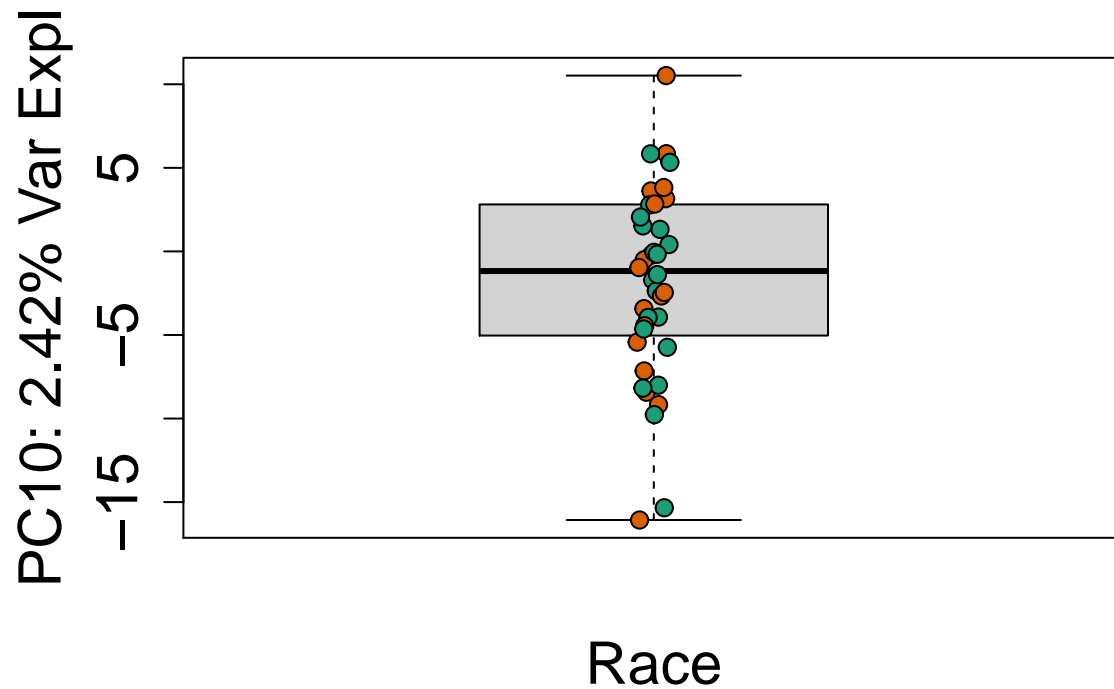












```
# Write plots to PDF
pdf(here("DE_analysis", "pdfs", "PCA_plotsExprs.pdf"), w=9)
generate_plots()
dev.off()
```

```
## pdf
## 2
```

1.6 Modeling

```
dge = DGEList(counts = assays(rse_gene)$counts,
              genes = rowData(rse_gene))
dge = calcNormFactors(dge)

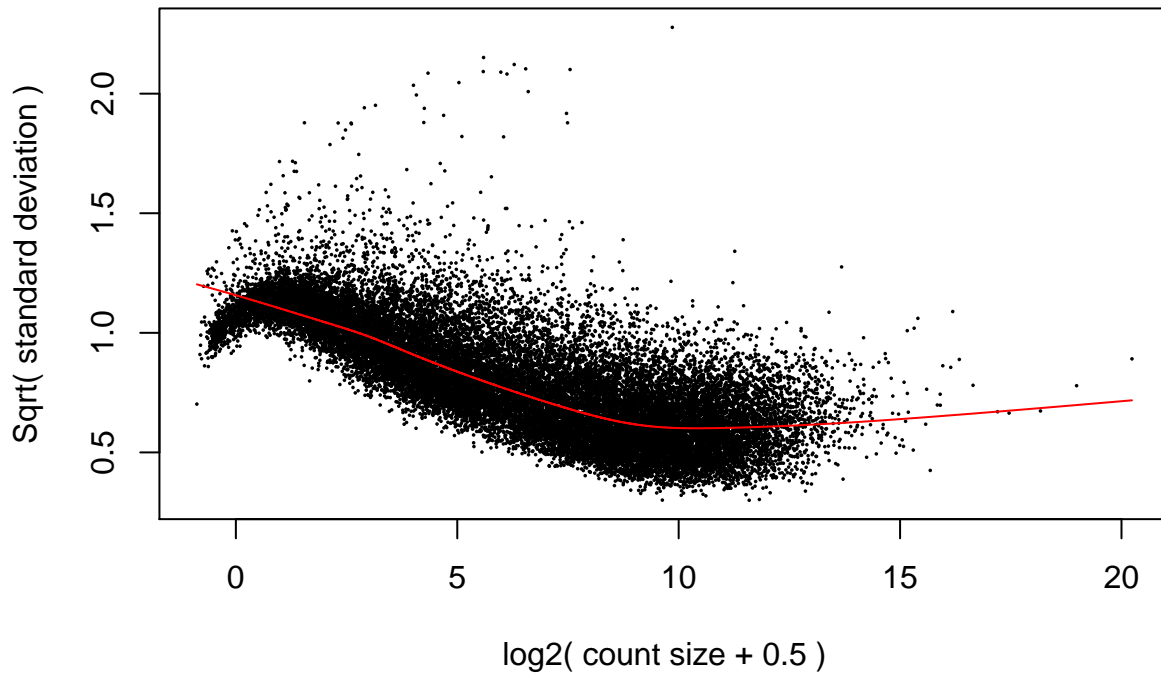
# Mean-variance
mod = model.matrix(~PrimaryDx + PC + BrainRegion,
                  data=colData(rse_gene))

vGene = invisible(voom(dge,mod,plot=TRUE))
```

```
## Coefficients not estimable: PrimaryDxOther
```

```
## Warning: Partial NA coefficients for 26708 probe(s)
```

voom: Mean–variance trend



```
# Also write mean-variance plot to PDF
pdf(file = "DE_analysis/pdfs/vGene.pdf")
invisible(voom(dge,mod,plot=TRUE))
```

```
## Coefficients not estimable: PrimaryDxOther
```

```
## Warning: Partial NA coefficients for 26708 probe(s)
```

```
dev.off()
```

```
## pdf
## 2
```

```
# Get duplicate correlation
gene_dupCorr = duplicateCorrelation(vGene$E, mod,
  block=colData(rse_gene)$SAMPLE_ID)
```

```
## Coefficients not estimable: PrimaryDxOther
```

```
save(gene_dupCorr, file = here("DE_analysis", "rdas", "gene_dupCorr_neurons.rda"))
```

```
# Fit linear model
fitGeneDupl = lmFit(vGene,
  correlation=gene_dupCorr$consensus.correlation,
  block=colData(rse_gene)$SAMPLEID)
```

```
## Coefficients not estimable: PrimaryDxOther
```

```
## Warning: Partial NA coefficients for 26708 probe(s)
```

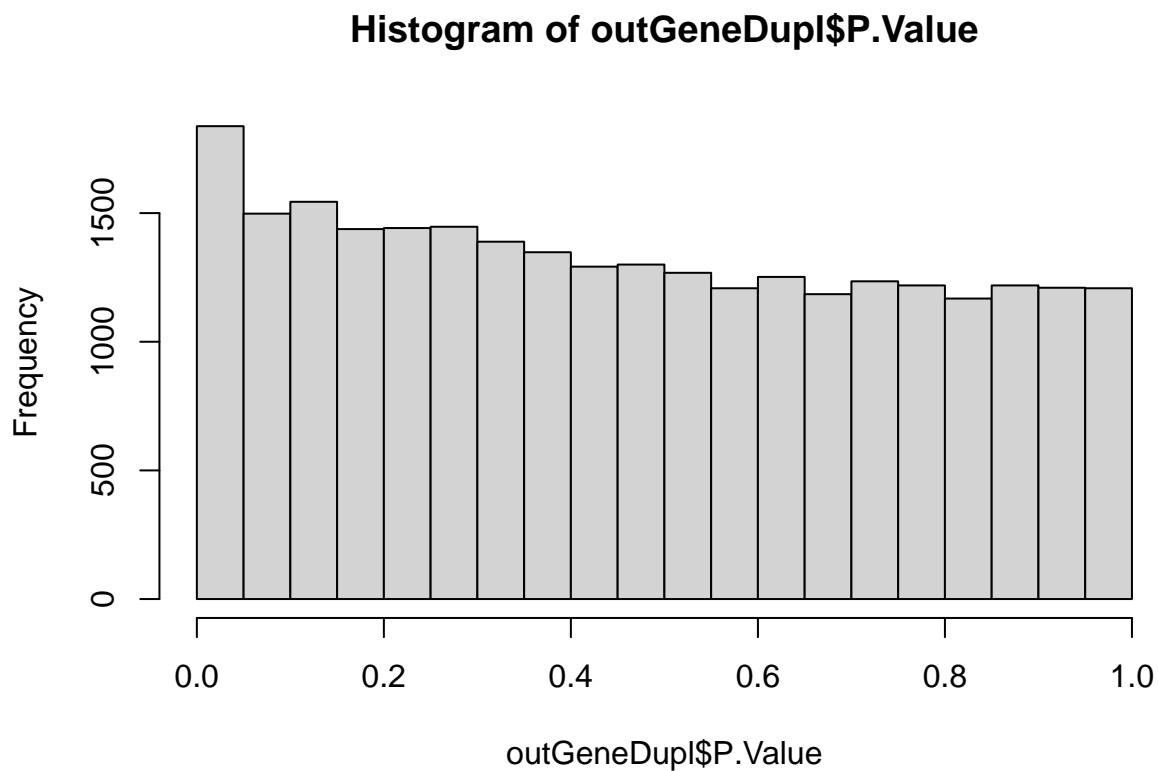
```
# Here we perform an empirical Bayesian calculation to obtain our significant genes  
ebGeneDupl = eBayes(fitGeneDupl)
```

```
## Warning in .ebayes(fit = fit, proportion = proportion, stdev.coef.lim = stdev.coef.lim, : Estimation  
## default value
```

```
outGeneDupl = topTable(ebGeneDupl,coef=2,  
  p.value = 1,number=nrow(rse_gene),sort="none")  
  
pdf(file = here("DE_analysis", "pdfs", "hist_pval.pdf"))  
hist(outGeneDupl$P.Value)  
dev.off()
```

```
## pdf  
## 2
```

```
hist(outGeneDupl$P.Value)
```



```
table(outGeneDupl$adj.P.Val < 0.05)
```

```
##
## FALSE TRUE
## 26707      1
```

```
table(outGeneDupl$adj.P.Val < 0.1)
```

```
##
## FALSE TRUE
## 26706      2
```

```
sigGeneDupl = topTable(ebGeneDupl,coef=2,
  p.value = 0.1,number=nrow(rse_gene))
```

```
sigGeneDupl[,c("Symbol", "logFC", "P.Value", "AveExpr")]
```

```
##
##          Symbol      logFC      P.Value  AveExpr
## ENSG00000269699.6    ZIM2 -1.1238665 1.448916e-06 0.8970203
## ENSG00000126368.6   NR1D1 -0.7637626 6.833687e-06 4.6128957
```

```
sigGeneDupl[sigGeneDupl$logFC > 0,c("Symbol", "logFC", "P.Value")]
```

```
## [1] Symbol logFC P.Value
## <0 rows> (or 0-length row.names)
```

```
sigGeneDupl[sigGeneDupl$logFC < 0,c("Symbol", "logFC", "P.Value")]
```

```
##
##          Symbol      logFC      P.Value
## ENSG00000269699.6    ZIM2 -1.1238665 1.448916e-06
## ENSG00000126368.6   NR1D1 -0.7637626 6.833687e-06
```

```
write.csv(outGeneDupl, file = here("DE_analysis", "tables", "de_stats_allExprs.csv"))
write.csv(sigGeneDupl, file = here("DE_analysis", "tables", "de_stats_fdr10_sorted.csv"))
```

1.7 Check plots

```
exprs = vGene$E[rownames(sigGeneDupl),]
```

```
# Group together code for displaying boxplots
```

```
generate_plots = function() {
  par(mar=c(8,6,4,2),cex.axis=1.8,cex.lab=1.8, cex.main=1.8)
  palette(brewer.pal(4,"Dark2"))
```

```
  for(i in 1:nrow(sigGeneDupl)) {
    yy = exprs[i,]
    boxplot(yy ~ rse_gene$PrimaryDx, outline=FALSE,
```

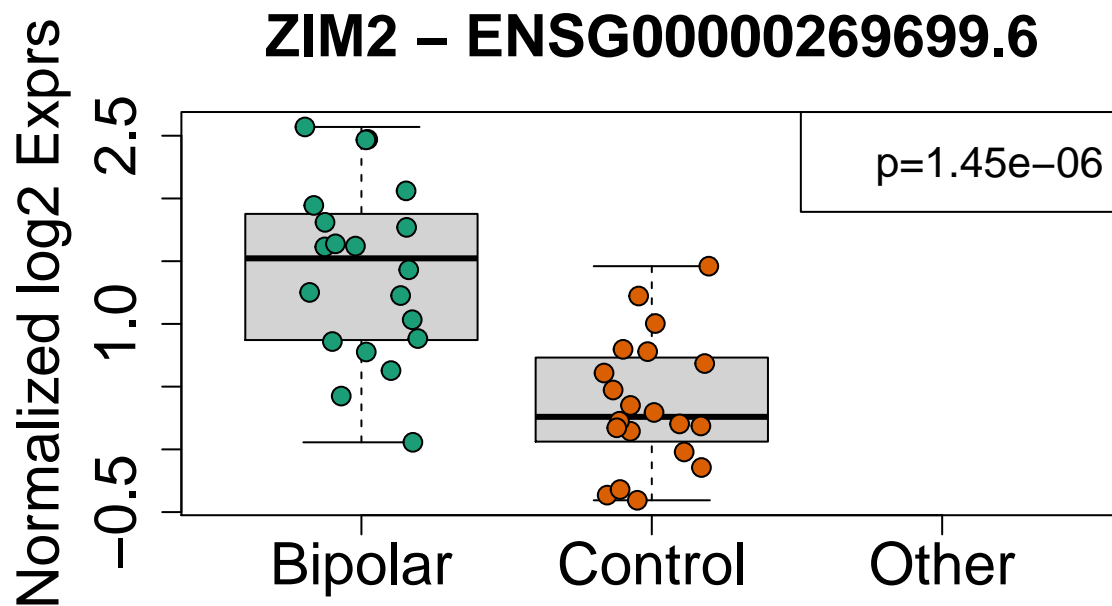


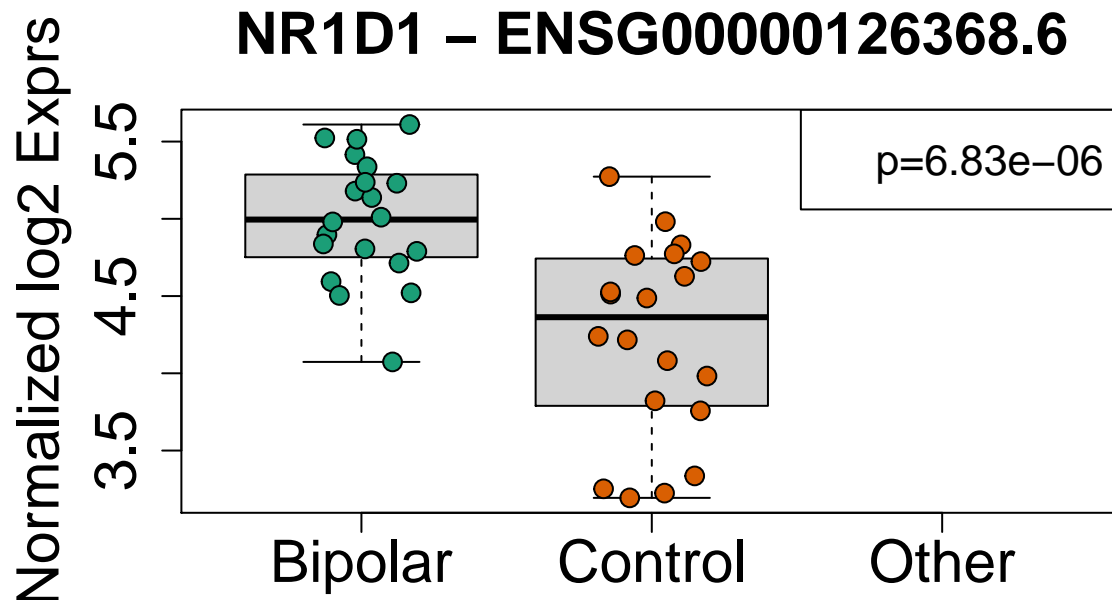
```

    ylim=range(yy), ylab="Normalized log2 Exprs", xlab="",
    main = paste(sigGeneDupl$Symbol[i], "-", sigGeneDupl$gencodeID[i]))
    points(yy ~ jitter(as.numeric(rse_gene$PrimaryDx)),
           pch = 21, bg= rse_gene$PrimaryDx,cex=1.3)
    ll = ifelse(sigGeneDupl$logFC[i] > 0, "topleft", "topright")
    legend(ll, paste0("p=", signif(sigGeneDupl$P.Value[i],3)), cex=1.3)
  }
}

# Show boxplots
generate_plots()

```





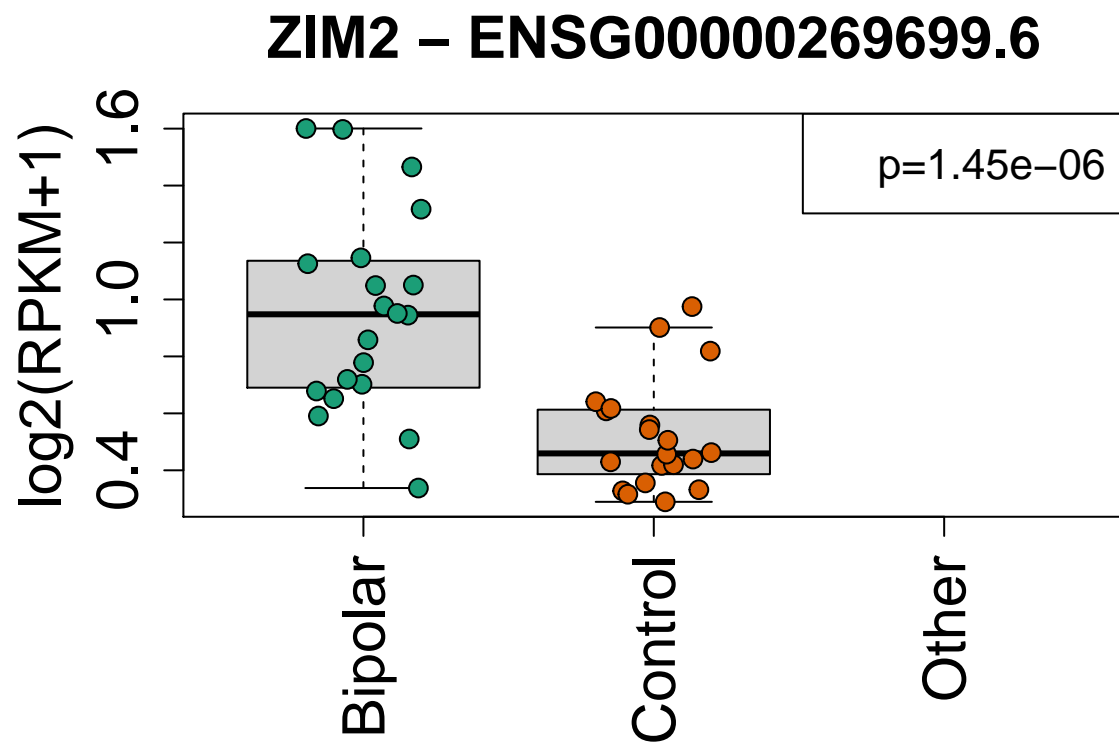
```
# Write plots to PDF
pdf(here("DE_analysis", "pdfs", "DE_boxplots_byDiagnosis.pdf"), w=10)
generate_plots()
dev.off()
```

```
## pdf
## 2
```

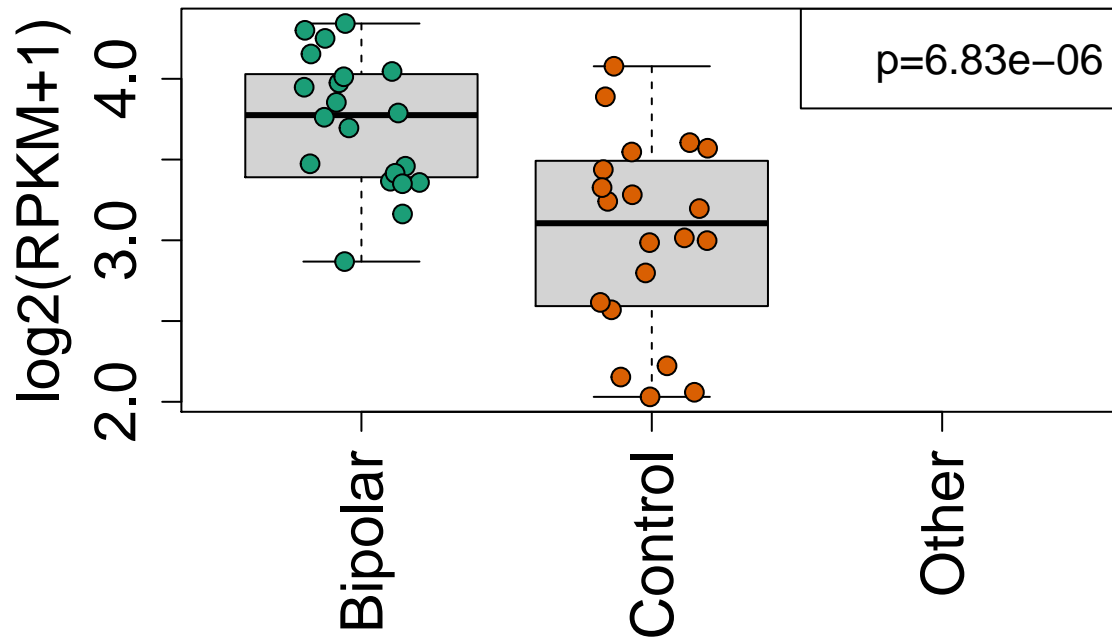
```
e = geneExprs[rownames(sigGeneDupl),]

generate_plots = function() {
  par(mar=c(8,6,4,2),cex.axis=1.8,cex.lab=1.8, cex.main=1.8)
  palette(brewer.pal(4,"Dark2"))
  for(i in 1:nrow(sigGeneDupl)) {
    yy = e[i,]
    boxplot(yy ~ rse_gene$PrimaryDx, las=3,outline=FALSE,
            ylim=range(yy), ylab="log2(RPKM+1)", xlab="",
            main = paste(sigGeneDupl$Symbol[i], "-", sigGeneDupl$gencodeID[i]))
    points(yy ~ jitter(as.numeric(rse_gene$PrimaryDx)),
           pch = 21, bg= rse_gene$PrimaryDx,cex=1.3)
    ll = ifelse(sigGeneDupl$logFC[i] > 0, "topleft", "topright")
    legend(ll, paste0("p=", signif(sigGeneDupl$P.Value[i],3)), cex=1.3)
  }
}
```

```
# Display plots  
generate_plots()
```



NR1D1 – ENSG00000126368.6



```
# Write the same plots to PDF
pdf(here("DE_analysis", "pdfs", "DE_boxplots_byGenome_log2RPKM.pdf"), w=10)
generate_plots()
dev.off()
```

```
## pdf
## 2
```

1.8 Gene ontology

clusterProfiler is a gene ontology package we will use to see if our genes are specifically differentially expressed in certain pathways.

```
# Get significant genes by sign
sigGene = outGeneDupl[outGeneDupl$P.Value < 0.005,]
sigGeneList = split(as.character(sigGene$EntrezID), sign(sigGene$logFC))
sigGeneList = lapply(sigGeneList, function(x) x[!is.na(x)])
geneUniverse = as.character(outGeneDupl$EntrezID)
geneUniverse = geneUniverse[!is.na(geneUniverse)]

# Do GO and KEGG
goBP_Adj <- compareCluster(sigGeneList, fun = "enrichGO",
  universe = geneUniverse, OrgDb = org.Hs.eg.db,
  ont = "BP", pAdjustMethod = "BH", pvalueCutoff = 1,
```

```

qvalueCutoff = 1, readable= TRUE)

goMF_Adj <- compareCluster(sigGeneList, fun = "enrichGO",
  universe = geneUniverse, OrgDb = org.Hs.eg.db,
  ont = "MF", pAdjustMethod = "BH", pvalueCutoff = 1,
  qvalueCutoff = 1, readable= TRUE)

goCC_Adj <- compareCluster(sigGeneList, fun = "enrichGO",
  universe = geneUniverse, OrgDb = org.Hs.eg.db,
  ont = "CC", pAdjustMethod = "BH", pvalueCutoff = 1,
  qvalueCutoff = 1, readable= TRUE)

kegg_Adj <- compareCluster(sigGeneList, fun = "enrichKEGG",
  universe = geneUniverse, pAdjustMethod = "BH",
  pvalueCutoff = 1, qvalueCutoff = 1)

save(goBP_Adj, goCC_Adj, goMF_Adj, kegg_Adj,
  file = here("DE_analysis", "rdas", "gene_set_objects_p005.rda"))

goList = list(BP = goBP_Adj, MF = goMF_Adj, CC = goCC_Adj, KEGG = kegg_Adj)
goDf = dplyr::bind_rows(lapply(goList, as.data.frame), .id = "Ontology")
goDf = goDf[order(goDf$pvalue),]

write.csv(goDf, file = here("DE_analysis", "tables", "geneSet_output.csv"), row.names=FALSE)

options(width=130)
goDf[goDf$p.adjust < 0.05, c(1:5,7)]

```

##	Ontology	Cluster	ID	Description
## 1311	BP	1	GO:0032640	tumor necrosis factor p
## 1312	BP	1	GO:0071706	tumor necrosis factor superfamily cytokine p
## 1313	BP	1	GO:0050865	regulation of cell a
## 1314	BP	1	GO:0032611	interleukin-1 beta p
## 1315	BP	1	GO:0032612	interleukin-1 p
## 1316	BP	1	GO:0032680	regulation of tumor necrosis factor p
## 1317	BP	1	GO:1903555	regulation of tumor necrosis factor superfamily cytokine p
## 1318	BP	1	GO:0032651	regulation of interleukin-1 beta p
## 1319	BP	1	GO:0050867	positive regulation of cell a
## 1320	BP	1	GO:0002521	leukocyte differ
## 1321	BP	1	GO:0002694	regulation of leukocyte a
## 1322	BP	1	GO:0032652	regulation of interleukin-1 p
## 1323	BP	1	GO:0009617	response to l
## 1324	BP	1	GO:0042742	defense response to l
## 1325	BP	1	GO:0071346	cellular response to interfe
## 1326	BP	1	GO:0050900	leukocyte r
## 3316	MF	1	GO:0045028	G protein-coupled purinergic nucleotide receptor
## 1327	BP	1	GO:0002687	positive regulation of leukocyte r
## 1328	BP	1	GO:0034341	response to interfe
## 3317	MF	1	GO:0046935	1-phosphatidylinositol-3-kinase regulator
## 1329	BP	1	GO:0031347	regulation of defense
## 1330	BP	1	GO:0002685	regulation of leukocyte r
## 1331	BP	1	GO:0071219	cellular response to molecule of bacteri
## 1332	BP	1	GO:0032760	positive regulation of tumor necrosis factor p

## 1333	BP	1	GO:1903975	regulation of glial cell m
## 1334	BP	1	GO:1903557	positive regulation of tumor necrosis factor superfamily cytokine p
## 1335	BP	1	GO:0032635	interleukin-6 p
## 1336	BP	1	GO:0006909	phag
## 1337	BP	1	GO:0032609	interferon-gamma p
## 1338	BP	1	GO:0042116	macrophage a
## 3318	MF	1	GO:0035014	phosphatidylinositol 3-kinase regulator
## 1339	BP	1	GO:0002696	positive regulation of leukocyte a
## 1340	BP	1	GO:0050702	interleukin-1 beta s
## 1341	BP	1	GO:0060216	definitive he
## 3319	MF	1	GO:0001614	purinergic nucleotide receptor
## 3320	MF	1	GO:0016502	nucleotide receptor
## 1342	BP	1	GO:0071216	cellular response to biotic
## 3321	MF	1	GO:0004930	G protein-coupled receptor
## 1343	BP	1	GO:0050701	interleukin-1 s
## 1344	BP	1	GO:0043032	positive regulation of macrophage a
## 1345	BP	1	GO:0050901	leukocyte tethering o
## 1346	BP	1	GO:0032729	positive regulation of interferon-gamma p
## 1347	BP	1	GO:0006911	phagocytosis, e
## 1348	BP	1	GO:0070098	chemokine-mediated signaling
## 1349	BP	1	GO:0007599	h
## 1350	BP	1	GO:0032691	negative regulation of interleukin-1 beta p
## 3137	MF	-1	GO:0070888	E-bo
## 1351	BP	1	GO:0032692	negative regulation of interleukin-1 p
## 1352	BP	1	GO:0001819	positive regulation of cytokine p
## 1353	BP	1	GO:0031349	positive regulation of defense
## 1354	BP	1	GO:0050766	positive regulation of phag
## 3871	KEGG	1	hsa04061	Viral protein interaction with cytokine and cytokine
## 1355	BP	1	GO:0043410	positive regulation of MAPK
## 1356	BP	1	GO:0071222	cellular response to lipopolys
## 1357	BP	1	GO:0099024	plasma membrane inv
## 1358	BP	1	GO:1990868	response to c
## 1359	BP	1	GO:1990869	cellular response to c
## 1360	BP	1	GO:0032637	interleukin-8 p
## 1361	BP	1	GO:0050729	positive regulation of inflammatory
## 1362	BP	1	GO:0050718	positive regulation of interleukin-1 beta s
## 1363	BP	1	GO:0032675	regulation of interleukin-6 p
## 1364	BP	1	GO:0035590	purinergic nucleotide receptor signaling
## 1365	BP	1	GO:0050716	positive regulation of interleukin-1 s
## 1366	BP	1	GO:0032103	positive regulation of response to external
## 1367	BP	1	GO:0002831	regulation of response to biotic
## 1368	BP	1	GO:0002275	myeloid cell activation involved in immune
## 1369	BP	1	GO:0030851	granulocyte differ
## 1370	BP	1	GO:2000403	positive regulation of lymphocyte m
## 1371	BP	1	GO:0002573	myeloid leukocyte differ
## 1372	BP	1	GO:0010324	membrane inv
## 1373	BP	1	GO:0061756	leukocyte adhesion to vascular endothe
## 1374	BP	1	GO:0032649	regulation of interferon-gamma p
## 3872	KEGG	1	hsa04060	Cytokine-cytokine receptor in
## 3322	MF	1	GO:0005126	cytokine recepto
## 1375	BP	1	GO:0032496	response to lipopolys
## 1376	BP	1	GO:1903532	positive regulation of secretion
## 1377	BP	1	GO:0072676	lymphocyte m
## 1378	BP	1	GO:0007159	leukocyte cell-cell

## 1379	BP	1 GO:0032755	positive regulation of interleukin-6 p
## 1380	BP	1 GO:0002237	response to molecule of bacteri
## 1381	BP	1 GO:0002683	negative regulation of immune system
## 1382	BP	1 GO:1905521	regulation of macrophage m
## 1383	BP	1 GO:0007596	blood co
## 1384	BP	1 GO:0050817	co
## 1385	BP	1 GO:0050706	regulation of interleukin-1 beta s
## 1386	BP	1 GO:0051047	positive regulation of s
## 1387	BP	1 GO:0042035	regulation of cytokine biosynthetic
## 1388	BP	1 GO:0050764	regulation of phag
## 1389	BP	1 GO:0001774	microglial cell a
## 1390	BP	1 GO:0002269	leukocyte activation involved in inflammatory
## 1391	BP	1 GO:0030225	macrophage differ
## 1392	BP	1 GO:0050704	regulation of interleukin-1 s
## 3323	MF	1 GO:0000978	RNA polymerase II cis-regulatory region sequence-specific DNA
## 1393	BP	1 GO:0030335	positive regulation of cell m
## 3324	MF	1 GO:0005525	GT
## 1394	BP	1 GO:0030099	myeloid cell differ
## 1395	BP	1 GO:0072604	interleukin-6 s
## 3325	MF	1 GO:0032550	purine ribonucleoside
## 3326	MF	1 GO:0001883	purine nucleoside
## 3327	MF	1 GO:0032549	ribonucleoside
## 1396	BP	1 GO:0008217	regulation of blood
## 1397	BP	1 GO:0042089	cytokine biosynthetic
## 1398	BP	1 GO:0042107	cytokine metabolic
## 3328	MF	1 GO:0000987	cis-regulatory region sequence-specific DNA
## 1399	BP	1 GO:1902105	regulation of leukocyte differ
## 1400	BP	1 GO:0045766	positive regulation of ang
## 1401	BP	1 GO:2000147	positive regulation of cell
## 1402	BP	1 GO:0002363	alpha-beta T cell lineage c
## 1403	BP	1 GO:0150078	positive regulation of neuroinflammatory
## 1404	BP	1 GO:0033003	regulation of mast cell a
## 3329	MF	1 GO:0001882	nucleoside
## 1405	BP	1 GO:0032731	positive regulation of interleukin-1 beta p
## 3330	MF	1 GO:0019001	guanyl nucleotide
## 3331	MF	1 GO:0032561	guanyl ribonucleotide
## 1406	BP	1 GO:0001818	negative regulation of cytokine p
## 1407	BP	1 GO:0048863	stem cell differ
## 1408	BP	1 GO:0043030	regulation of macrophage a
## 1409	BP	1 GO:0002697	regulation of immune effecto
## 1410	BP	1 GO:0040017	positive regulation of l
## 1411	BP	1 GO:0043369	CD4-positive or CD8-positive, alpha-beta T cell lineage c
## 1412	BP	1 GO:0098883	synaps
## 1413	BP	1 GO:0051272	positive regulation of cellular component
## 3332	MF	1 GO:0070888	E-bo
## 1414	BP	1 GO:0097529	myeloid leukocyte m
## 1415	BP	1 GO:0043300	regulation of leukocyte degr
## 1416	BP	1 GO:0002244	hematopoietic progenitor cell differ
## 1417	BP	1 GO:0002703	regulation of leukocyte mediated
## 1418	BP	1 GO:0070374	positive regulation of ERK1 and ERK2
## 3333	MF	1 GO:0048020	CCR chemokine recepto
## 1419	BP	1 GO:1903708	positive regulation of her
## 1420	BP	1 GO:0032732	positive regulation of interleukin-1 p
## 1421	BP	1 GO:1905517	macrophage m

## 1422	BP	1 GO:0070372	regulation of ERK1 and ERK2
## 1423	BP	1 GO:1903037	regulation of leukocyte cell-cell adhesion
## 1424	BP	1 GO:0071692	protein localization to extracellular space
## 1425	BP	1 GO:0045088	regulation of innate immune response
## 1426	BP	1 GO:0061900	glial cell development
## 1427	BP	1 GO:0010919	regulation of inositol phosphate biosynthesis
## 1428	BP	1 GO:0033004	negative regulation of mast cell activation
## 1429	BP	1 GO:0060100	positive regulation of phagocytosis, endocytosis
## 1430	BP	1 GO:1905155	positive regulation of membrane invagination
## 1431	BP	1 GO:0050878	regulation of body fluid homeostasis
## 1432	BP	1 GO:0002761	regulation of myeloid leukocyte differentiation
## 1433	BP	1 GO:1904018	positive regulation of vasculature development
## 1434	BP	1 GO:0008347	glial cell development
## 1435	BP	1 GO:0043551	regulation of phosphatidylinositol 3-kinase activity
## 1436	BP	1 GO:0072678	T cell lineage commitment
## 3334	MF	1 GO:0019955	cytokine production
## 1437	BP	1 GO:0001562	response to peptide stimulus
## 1438	BP	1 GO:0042832	defense response to peptide
## 1439	BP	1 GO:0060099	regulation of phagocytosis, endocytosis
## 1440	BP	1 GO:1903978	regulation of microglial cell activation
## 1441	BP	1 GO:2000401	regulation of lymphocyte migration
## 1442	BP	1 GO:0043270	positive regulation of ion transport
## 1443	BP	1 GO:0042108	positive regulation of cytokine biosynthesis
## 1444	BP	1 GO:0070371	ERK1 and ERK2
## 1445	BP	1 GO:0002790	peptide hormone
## 1446	BP	1 GO:0002360	T cell lineage commitment
## 1447	BP	1 GO:0042104	positive regulation of activated T cell proliferation
## 1448	BP	1 GO:1905153	regulation of membrane invagination
## 1449	BP	1 GO:0002886	regulation of myeloid leukocyte mediated immunity
## 1450	BP	1 GO:0045765	regulation of angiogenesis
## 1451	BP	1 GO:0032677	regulation of interleukin-8 production
## 1452	BP	1 GO:0045123	cellular extracellular matrix
## 1453	BP	1 GO:0050727	regulation of inflammatory response
## 1454	BP	1 GO:0002577	regulation of antigen processing and presentation
## 1455	BP	1 GO:0150146	cell junction disassembly
## 1456	BP	1 GO:0006954	inflammatory response
## 1457	BP	1 GO:0031663	lipopolysaccharide-mediated signaling pathway
##	pvalue		
## 1311	4.442846e-08		
## 1312	5.955807e-08		
## 1313	2.042035e-07		
## 1314	2.762117e-07		
## 1315	4.846558e-07		
## 1316	5.291141e-07		
## 1317	6.457953e-07		
## 1318	2.297132e-06		
## 1319	2.349005e-06		
## 1320	3.605350e-06		
## 1321	3.605350e-06		
## 1322	3.637541e-06		
## 1323	6.604984e-06		
## 1324	9.663583e-06		
## 1325	2.188166e-05		
## 1326	2.555242e-05		

3316 3.086309e-05
1327 3.901699e-05
1328 3.938836e-05
3317 4.223712e-05
1329 4.302992e-05
1330 4.665008e-05
1331 4.862740e-05
1332 5.573978e-05
1333 6.417224e-05
1334 6.438378e-05
1335 6.633435e-05
1336 7.418961e-05
1337 7.924753e-05
1338 8.474696e-05
3318 9.186960e-05
1339 9.207168e-05
1340 9.843140e-05
1341 1.011745e-04
3319 1.142974e-04
3320 1.142974e-04
1342 1.169372e-04
3321 1.277876e-04
1343 1.320556e-04
1344 1.498475e-04
1345 1.498475e-04
1346 1.586476e-04
1347 1.733017e-04
1348 1.733017e-04
1349 2.032086e-04
1350 2.116152e-04
3137 2.187705e-04
1351 2.478391e-04
1352 2.514522e-04
1353 2.564692e-04
1354 2.825185e-04
3871 2.923628e-04
1355 2.986591e-04
1356 3.212669e-04
1357 3.279249e-04
1358 3.279249e-04
1359 3.279249e-04
1360 3.783360e-04
1361 3.798024e-04
1362 4.321639e-04
1363 4.766011e-04
1364 4.888890e-04
1365 4.888890e-04
1366 4.971077e-04
1367 5.110894e-04
1368 5.390682e-04
1369 5.501749e-04
1370 5.501749e-04
1371 5.551129e-04
1372 5.626928e-04

1373 6.161675e-04
1374 6.754461e-04
3872 7.205888e-04
3322 7.359591e-04
1375 7.424577e-04
1376 7.846042e-04
1377 8.033448e-04
1378 8.182442e-04
1379 8.495298e-04
1380 9.655090e-04
1381 9.791509e-04
1382 1.021620e-03
1383 1.034691e-03
1384 1.082942e-03
1385 1.118743e-03
1386 1.143604e-03
1387 1.166745e-03
1388 1.166745e-03
1389 1.330039e-03
1390 1.330039e-03
1391 1.330039e-03
1392 1.330039e-03
3323 1.337238e-03
1393 1.367487e-03
3324 1.384455e-03
1394 1.431051e-03
1395 1.444451e-03
3325 1.493956e-03
3326 1.580520e-03
3327 1.580520e-03
1396 1.596835e-03
1397 1.632240e-03
1398 1.632240e-03
3328 1.632513e-03
1399 1.635117e-03
1400 1.649303e-03
1401 1.654766e-03
1402 1.677685e-03
1403 1.677685e-03
1404 1.691385e-03
3329 1.798052e-03
1405 1.824131e-03
3330 1.830975e-03
3331 1.830975e-03
1406 1.942791e-03
1407 1.942791e-03
1408 1.963208e-03
1409 2.003158e-03
1410 2.018912e-03
1411 2.042202e-03
1412 2.042202e-03
1413 2.049387e-03
3332 2.051128e-03
1414 2.052809e-03

```
## 1415 2.108721e-03
## 1416 2.115847e-03
## 1417 2.180288e-03
## 1418 2.180288e-03
## 3333 2.213958e-03
## 1419 2.246151e-03
## 1420 2.260772e-03
## 1421 2.260772e-03
## 1422 2.292524e-03
## 1423 2.346125e-03
## 1424 2.374893e-03
## 1425 2.400666e-03
## 1426 2.419462e-03
## 1427 2.440725e-03
## 1428 2.440725e-03
## 1429 2.440725e-03
## 1430 2.440725e-03
## 1431 2.485301e-03
## 1432 2.602808e-03
## 1433 2.748562e-03
## 1434 2.757143e-03
## 1435 2.757143e-03
## 1436 2.757143e-03
## 3334 2.788894e-03
## 1437 2.872826e-03
## 1438 2.872826e-03
## 1439 2.872826e-03
## 1440 2.872826e-03
## 1441 2.936322e-03
## 1442 3.000153e-03
## 1443 3.122514e-03
## 1444 3.132401e-03
## 1445 3.149398e-03
## 1446 3.338079e-03
## 1447 3.338079e-03
## 1448 3.338079e-03
## 1449 3.516285e-03
## 1450 3.629843e-03
## 1451 3.724031e-03
## 1452 3.724031e-03
## 1453 3.782205e-03
## 1454 3.836064e-03
## 1455 3.836064e-03
## 1456 3.902893e-03
## 1457 3.939125e-03
```

1.9 Visualize differentially expressed genes

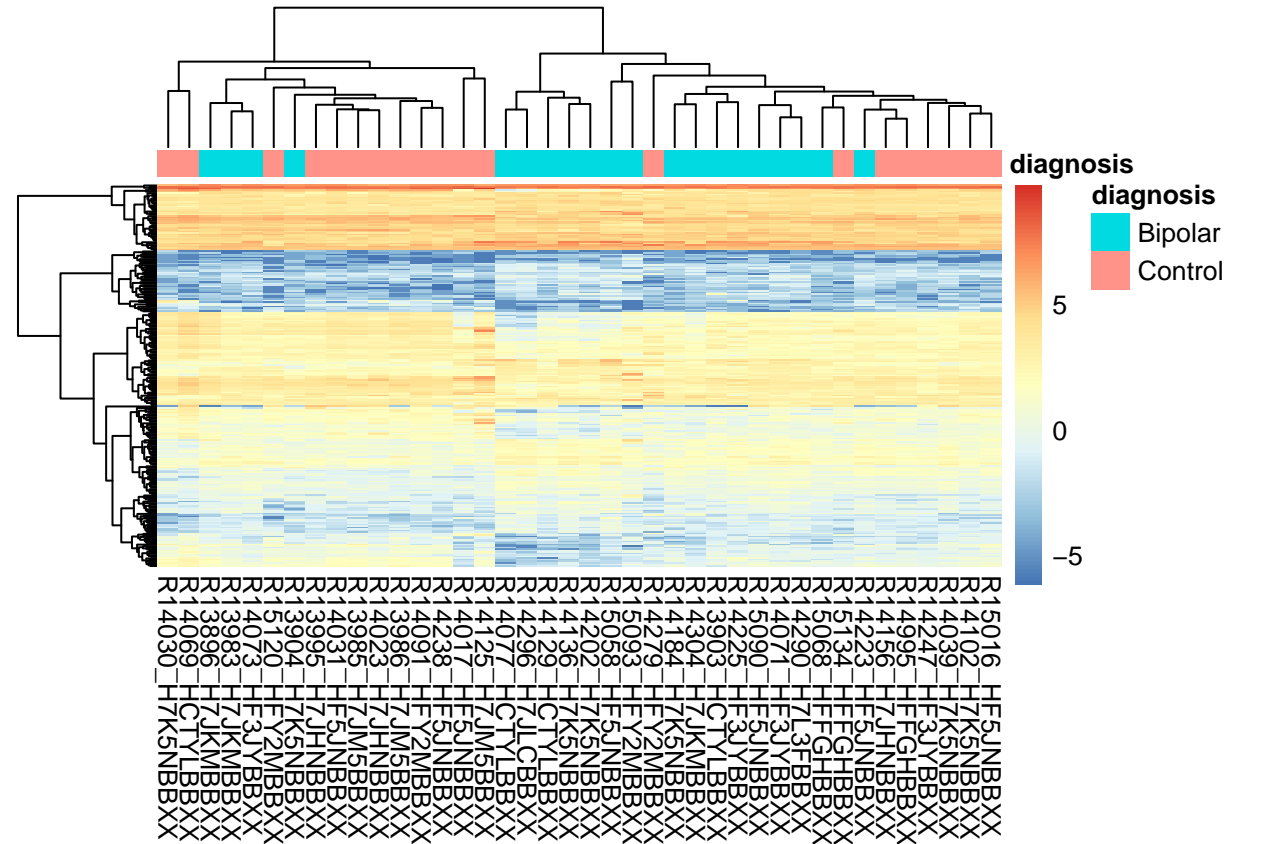
Here we visualize DEGs with a heatmap.

```
exprs_heatmap = vGene$E[rownames(sigGene),]

df <- as.data.frame(colData(rse_gene)[,c("PrimaryDx")])
```

```
rownames(df) <- colnames(exprs_heatmap)
colnames(df) <- "diagnosis"

# Display heatmap
pheatmap(exprs_heatmap, cluster_rows=TRUE, show_rownames=FALSE,
          cluster_cols=TRUE, annotation_col=df)
```



```
# Write heatmap to PDF
pdf(file=here("DE_analysis", "pdfs", "de_heatmap.pdf"))
pheatmap(exprs_heatmap, cluster_rows=TRUE, show_rownames=FALSE,
          cluster_cols=TRUE, annotation_col=df)
dev.off()
```

```
## pdf
## 3
```