GO\_term\_analysis

#LOAD PACKAGES

library('topGO') #for enrichment analysis  
library('ggplot2') #for plotting  
library("viridis") #for colours on graphs  
library("forcats") #for reordering bar graphs in fisher p-value order  
library('data.table') #for loading large files (i.e. GO annotations)  
library("VennDiagram") #for plotting venn diagrams  
library('WriteXLS') #for saving final output tables as XLS file

#TOPGO INFO

#note possible tests  
# Tests based on gene counts. This is the most popular family of tests, given that it only requires the presence of a list of interesting genes and nothing more. Tests like Fisher's exact test, Hypegeometric test and binomial test belong to this family. Draghici et al. (2006)  
  
# Tests based on gene scores or gene ranks. It includes Kolmogorov-Smirnov like tests (also known as GSEA), Gentleman's Category, t-test, etc. Ackermann and Strimmer (2009)  
  
# Tests based on gene expression. Tests like Goeman's globaltest or GlobalAncova separates from the others since they work directly on the expression matrix. Goeman and Bühlmann (2007)  
  
#https://www.nature.com/articles/s41477-018-0172-3 = used fisher and weight01 algorithm  
#https://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-020-06806-5 = goseq  
#https://journals.plos.org/plosgenetics/article?id=10.1371/journal.pgen.1008398#pgen.1008398.s001 = used fisher and weight01 algorithm, gives R script  
#https://onlinelibrary.wiley.com/doi/10.1002/evl3.269 = used fisher and weight algorithm

#GO DATABASE + DATA

#load GO database  
GO\_database <- readMappings(file = "GO\_analysis/R\_topGO/trinotate\_GO\_terms.tsv", sep = "\t", IDsep = ",")  
head(GO\_database, 1) #view first gene + associated GO terms

## $Qrob\_T0000010.2  
## [1] "GO:0005575" "GO:0005634" "GO:0043226" "GO:0043227" "GO:0043229"  
## [6] "GO:0043231" "GO:0044424" "GO:0044464"

#load input .Rdata from Deseq analysis  
load(file = "GO\_analysis/R\_topGO/Initial\_GO\_input\_data\_fc1.RData")

#MOCK 0DPI

#BABA  
  
#subset sig DEGs for 0dpi  
sig\_DEGs\_0dpi\_mock\_BABA\_vs\_mock\_Water <- subset(DEGs\_0dpi\_BABA, DEGs\_0dpi\_BABA$padj <= 0.05)   
  
#extract row names of sig DEGs for each time-point  
names\_sig\_DEGs\_0dpi\_mock\_BABA\_vs\_mock\_Water <- row.names(sig\_DEGs\_0dpi\_mock\_BABA\_vs\_mock\_Water)  
  
#set up vector to store genes of interest for GO analysis  
mock\_BABA\_0dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
mock\_BABA\_0dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_sig\_DEGs\_0dpi\_mock\_BABA\_vs\_mock\_Water))  
  
#add genes names to factor  
names(mock\_BABA\_0dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#SA  
  
#subset sig DEGs for 0dpi  
sig\_DEGs\_0dpi\_mock\_SA\_vs\_mock\_Water <- subset(DEGs\_0dpi\_SA, DEGs\_0dpi\_SA$padj <= 0.05)   
  
#extract row names of sig DEGs for each time-point  
names\_sig\_DEGs\_0dpi\_mock\_SA\_vs\_mock\_Water <- row.names(sig\_DEGs\_0dpi\_mock\_SA\_vs\_mock\_Water)  
  
#set up vector to store genes of interest for GO analysis  
mock\_SA\_0dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
mock\_SA\_0dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_sig\_DEGs\_0dpi\_mock\_SA\_vs\_mock\_Water))  
  
#add genes names to factor  
names(mock\_SA\_0dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#JA  
  
#subset sig DEGs for 0dpi  
sig\_DEGs\_0dpi\_mock\_JA\_vs\_mock\_Water <- subset(DEGs\_0dpi\_JA, DEGs\_0dpi\_JA$padj <= 0.05)   
  
#extract row names of sig DEGs for each time-point  
names\_sig\_DEGs\_0dpi\_mock\_JA\_vs\_mock\_Water <- row.names(sig\_DEGs\_0dpi\_mock\_JA\_vs\_mock\_Water)  
  
#set up vector to store genes of interest for GO analysis  
mock\_JA\_0dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
mock\_JA\_0dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_sig\_DEGs\_0dpi\_mock\_JA\_vs\_mock\_Water))  
  
#add genes names to factor  
names(mock\_JA\_0dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)

##———BABA mock 0dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_BABA\_mock\_0dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = mock\_BABA\_0dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_BABA\_mock\_0dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_BABA\_mock\_0dpi\_GO\_results <- GenTable(BP\_BABA\_mock\_0dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_BABA\_mock\_0dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_BABA\_mock\_0dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = mock\_BABA\_0dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_BABA\_mock\_0dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_BABA\_mock\_0dpi\_GO\_results <- GenTable(MF\_BABA\_mock\_0dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_BABA\_mock\_0dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_BABA\_mock\_0dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = mock\_BABA\_0dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_BABA\_mock\_0dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_BABA\_mock\_0dpi\_GO\_results <- GenTable(CC\_BABA\_mock\_0dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_BABA\_mock\_0dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_BABA\_mock\_0dpi\_GO\_results$fisher <- as.numeric(BP\_BABA\_mock\_0dpi\_GO\_results$fisher)  
class(BP\_BABA\_mock\_0dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_BABA\_mock\_0dpi\_GO\_results\_top\_25 <- BP\_BABA\_mock\_0dpi\_GO\_results[1:25,]   
BP\_BABA\_mock\_0dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_BABA\_mock\_0dpi\_GO\_results\_plot\_data <- subset(BP\_BABA\_mock\_0dpi\_GO\_results\_top\_25, BP\_BABA\_mock\_0dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_BABA\_mock\_0dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_BABA\_mock\_0dpi\_GO\_graph <-   
 ggplot(data = BP\_BABA\_mock\_0dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,400), breaks = seq(0,400,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_BABA\_mock\_0dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_BABA\_mock\_0dpi\_GO\_results$fisher <- as.numeric(MF\_BABA\_mock\_0dpi\_GO\_results$fisher)  
class(MF\_BABA\_mock\_0dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_BABA\_mock\_0dpi\_GO\_results\_top\_25 <- MF\_BABA\_mock\_0dpi\_GO\_results[1:25,]   
MF\_BABA\_mock\_0dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_BABA\_mock\_0dpi\_GO\_results\_plot\_data <- subset(MF\_BABA\_mock\_0dpi\_GO\_results\_top\_25, MF\_BABA\_mock\_0dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_BABA\_mock\_0dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_BABA\_mock\_0dpi\_GO\_graph <-   
 ggplot(data = MF\_BABA\_mock\_0dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,400), breaks = seq(0,400,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_BABA\_mock\_0dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_BABA\_mock\_0dpi\_GO\_results$fisher <- as.numeric(CC\_BABA\_mock\_0dpi\_GO\_results$fisher)  
class(CC\_BABA\_mock\_0dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_BABA\_mock\_0dpi\_GO\_results\_top\_25 <- CC\_BABA\_mock\_0dpi\_GO\_results[1:25,]   
CC\_BABA\_mock\_0dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_BABA\_mock\_0dpi\_GO\_results\_plot\_data <- subset(CC\_BABA\_mock\_0dpi\_GO\_results\_top\_25, CC\_BABA\_mock\_0dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_BABA\_mock\_0dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_BABA\_mock\_0dpi\_GO\_graph <-   
 ggplot(data = CC\_BABA\_mock\_0dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,400), breaks = seq(0,400,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_BABA\_mock\_0dpi\_GO\_graph

#ARRANGE AND SAVE THE THREE PLOTS  
  
GO\_terms\_BABA\_mock\_0dpi <- cowplot::plot\_grid(BP\_BABA\_mock\_0dpi\_GO\_graph, MF\_BABA\_mock\_0dpi\_GO\_graph, CC\_BABA\_mock\_0dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")  
GO\_terms\_BABA\_mock\_0dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_BABA\_mock\_0dpi.pdf", plot = GO\_terms\_BABA\_mock\_0dpi, width=20, height=20, dpi=300)

##———SA mock 0dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_SA\_mock\_0dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = mock\_SA\_0dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_SA\_mock\_0dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_SA\_mock\_0dpi\_GO\_results <- GenTable(BP\_SA\_mock\_0dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_SA\_mock\_0dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_SA\_mock\_0dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = mock\_SA\_0dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_SA\_mock\_0dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_SA\_mock\_0dpi\_GO\_results <- GenTable(MF\_SA\_mock\_0dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_SA\_mock\_0dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_SA\_mock\_0dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = mock\_SA\_0dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_SA\_mock\_0dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_SA\_mock\_0dpi\_GO\_results <- GenTable(CC\_SA\_mock\_0dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_SA\_mock\_0dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_SA\_mock\_0dpi\_GO\_results$fisher <- as.numeric(BP\_SA\_mock\_0dpi\_GO\_results$fisher)  
class(BP\_SA\_mock\_0dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_SA\_mock\_0dpi\_GO\_results\_top\_25 <- BP\_SA\_mock\_0dpi\_GO\_results[1:25,]   
BP\_SA\_mock\_0dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_SA\_mock\_0dpi\_GO\_results\_plot\_data <- subset(BP\_SA\_mock\_0dpi\_GO\_results\_top\_25, BP\_SA\_mock\_0dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_SA\_mock\_0dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_SA\_mock\_0dpi\_GO\_graph <-   
 ggplot(data = BP\_SA\_mock\_0dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,380), breaks = seq(0,380,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_SA\_mock\_0dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_SA\_mock\_0dpi\_GO\_results$fisher <- as.numeric(MF\_SA\_mock\_0dpi\_GO\_results$fisher)  
class(MF\_SA\_mock\_0dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_SA\_mock\_0dpi\_GO\_results\_top\_25 <- MF\_SA\_mock\_0dpi\_GO\_results[1:25,]   
MF\_SA\_mock\_0dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_SA\_mock\_0dpi\_GO\_results\_plot\_data <- subset(MF\_SA\_mock\_0dpi\_GO\_results\_top\_25, MF\_SA\_mock\_0dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_SA\_mock\_0dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_SA\_mock\_0dpi\_GO\_graph <-   
 ggplot(data = MF\_SA\_mock\_0dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,380), breaks = seq(0,380,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_SA\_mock\_0dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_SA\_mock\_0dpi\_GO\_results$fisher <- as.numeric(CC\_SA\_mock\_0dpi\_GO\_results$fisher)  
class(CC\_SA\_mock\_0dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_SA\_mock\_0dpi\_GO\_results\_top\_25 <- CC\_SA\_mock\_0dpi\_GO\_results[1:25,]   
CC\_SA\_mock\_0dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_SA\_mock\_0dpi\_GO\_results\_plot\_data <- subset(CC\_SA\_mock\_0dpi\_GO\_results\_top\_25, CC\_SA\_mock\_0dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_SA\_mock\_0dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_SA\_mock\_0dpi\_GO\_graph <-   
 ggplot(data = CC\_SA\_mock\_0dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,380), breaks = seq(0,380,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_SA\_mock\_0dpi\_GO\_graph

#arrange plots and save  
GO\_terms\_SA\_mock\_0dpi <- cowplot::plot\_grid(BP\_SA\_mock\_0dpi\_GO\_graph, MF\_SA\_mock\_0dpi\_GO\_graph, CC\_SA\_mock\_0dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")  
GO\_terms\_SA\_mock\_0dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_SA\_mock\_0dpi.pdf", plot = GO\_terms\_SA\_mock\_0dpi, width=20, height=20, dpi=300)

##———JA mock 0dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_JA\_mock\_0dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = mock\_JA\_0dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_JA\_mock\_0dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_JA\_mock\_0dpi\_GO\_results <- GenTable(BP\_JA\_mock\_0dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_JA\_mock\_0dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_JA\_mock\_0dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = mock\_JA\_0dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_JA\_mock\_0dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_JA\_mock\_0dpi\_GO\_results <- GenTable(MF\_JA\_mock\_0dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_JA\_mock\_0dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_JA\_mock\_0dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = mock\_JA\_0dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_JA\_mock\_0dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_JA\_mock\_0dpi\_GO\_results <- GenTable(CC\_JA\_mock\_0dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_JA\_mock\_0dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_JA\_mock\_0dpi\_GO\_results$fisher <- as.numeric(BP\_JA\_mock\_0dpi\_GO\_results$fisher)  
class(BP\_JA\_mock\_0dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_JA\_mock\_0dpi\_GO\_results\_top\_25 <- BP\_JA\_mock\_0dpi\_GO\_results[1:25,]   
BP\_JA\_mock\_0dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_JA\_mock\_0dpi\_GO\_results\_plot\_data <- subset(BP\_JA\_mock\_0dpi\_GO\_results\_top\_25, BP\_JA\_mock\_0dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_JA\_mock\_0dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_JA\_mock\_0dpi\_GO\_graph <-   
 ggplot(data = BP\_JA\_mock\_0dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,300), breaks = seq(0,300,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_JA\_mock\_0dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_JA\_mock\_0dpi\_GO\_results$fisher <- as.numeric(MF\_JA\_mock\_0dpi\_GO\_results$fisher)  
class(MF\_JA\_mock\_0dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_JA\_mock\_0dpi\_GO\_results\_top\_25 <- MF\_JA\_mock\_0dpi\_GO\_results[1:25,]   
MF\_JA\_mock\_0dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_JA\_mock\_0dpi\_GO\_results\_plot\_data <- subset(MF\_JA\_mock\_0dpi\_GO\_results\_top\_25, MF\_JA\_mock\_0dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_JA\_mock\_0dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_JA\_mock\_0dpi\_GO\_graph <-   
 ggplot(data = MF\_JA\_mock\_0dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,300), breaks = seq(0,300,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_JA\_mock\_0dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_JA\_mock\_0dpi\_GO\_results$fisher <- as.numeric(CC\_JA\_mock\_0dpi\_GO\_results$fisher)  
class(CC\_JA\_mock\_0dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_JA\_mock\_0dpi\_GO\_results\_top\_25 <- CC\_JA\_mock\_0dpi\_GO\_results[1:25,]   
CC\_JA\_mock\_0dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_JA\_mock\_0dpi\_GO\_results\_plot\_data <- subset(CC\_JA\_mock\_0dpi\_GO\_results\_top\_25, CC\_JA\_mock\_0dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_JA\_mock\_0dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_JA\_mock\_0dpi\_GO\_graph <-   
 ggplot(data = CC\_JA\_mock\_0dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,300), breaks = seq(0,300,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_JA\_mock\_0dpi\_GO\_graph

#arrange plots and save  
GO\_terms\_JA\_mock\_0dpi <- cowplot::plot\_grid(BP\_JA\_mock\_0dpi\_GO\_graph, MF\_JA\_mock\_0dpi\_GO\_graph, CC\_JA\_mock\_0dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")  
GO\_terms\_JA\_mock\_0dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_JA\_mock\_0dpi.pdf", plot = GO\_terms\_JA\_mock\_0dpi, width=20, height=20, dpi=300)

#PRIMED 1DPI

#BABA  
#extract row names of primed DEGs  
names\_BABA\_primed\_all\_1dpi <- row.names(BABA\_primed\_all\_1dpi)  
  
#set up vector to store genes of interest for GO analysis  
BABA\_primed\_1dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data))) #can just use zero dpi raw gene data as row/gene names are the same for all three timepoints  
  
#set genes of interest to 1 and others to 0  
BABA\_primed\_1dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_BABA\_primed\_all\_1dpi))  
  
#add genes names to factor  
names(BABA\_primed\_1dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#SA  
#extract row names of primed DEGs  
names\_SA\_primed\_all\_1dpi <- row.names(SA\_primed\_all\_1dpi)  
  
#set up vector to store genes of interest for GO analysis  
SA\_primed\_1dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
SA\_primed\_1dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_SA\_primed\_all\_1dpi))  
  
#add genes names to factor  
names(SA\_primed\_1dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#JA  
#extract row names of primed DEGs  
names\_JA\_primed\_all\_1dpi <- row.names(JA\_primed\_all\_1dpi)  
  
#set up vector to store genes of interest for GO analysis  
JA\_primed\_1dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
JA\_primed\_1dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_JA\_primed\_all\_1dpi))  
  
#add genes names to factor  
names(JA\_primed\_1dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)

##———BABA Primed 1dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_BABA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = BABA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_BABA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_BABA\_primed\_1dpi\_GO\_results <- GenTable(BP\_BABA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_BABA\_primed\_1dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_BABA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = BABA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_BABA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_BABA\_primed\_1dpi\_GO\_results <- GenTable(MF\_BABA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_BABA\_primed\_1dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_BABA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = BABA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_BABA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_BABA\_primed\_1dpi\_GO\_results <- GenTable(CC\_BABA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_BABA\_primed\_1dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_BABA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(BP\_BABA\_primed\_1dpi\_GO\_results$fisher)  
class(BP\_BABA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_BABA\_primed\_1dpi\_GO\_results\_top\_25 <- BP\_BABA\_primed\_1dpi\_GO\_results[1:25,]   
BP\_BABA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(BP\_BABA\_primed\_1dpi\_GO\_results\_top\_25, BP\_BABA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_BABA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = BP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,300), breaks = seq(0,300,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_BABA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_BABA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(MF\_BABA\_primed\_1dpi\_GO\_results$fisher)  
class(MF\_BABA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_BABA\_primed\_1dpi\_GO\_results\_top\_25 <- MF\_BABA\_primed\_1dpi\_GO\_results[1:25,]   
MF\_BABA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_BABA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(MF\_BABA\_primed\_1dpi\_GO\_results\_top\_25, MF\_BABA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_BABA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_BABA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = MF\_BABA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,300), breaks = seq(0,300,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_BABA\_primed\_1dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_BABA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(CC\_BABA\_primed\_1dpi\_GO\_results$fisher)  
class(CC\_BABA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_BABA\_primed\_1dpi\_GO\_results\_top\_25 <- CC\_BABA\_primed\_1dpi\_GO\_results[1:25,]   
CC\_BABA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_BABA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(CC\_BABA\_primed\_1dpi\_GO\_results\_top\_25, CC\_BABA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_BABA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_BABA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = CC\_BABA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,300), breaks = seq(0,300,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_BABA\_primed\_1dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_BABA\_primed\_1dpi <- cowplot::plot\_grid(BP\_BABA\_primed\_1dpi\_GO\_graph, MF\_BABA\_primed\_1dpi\_GO\_graph, CC\_BABA\_primed\_1dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_BABA\_primed\_1dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_BABA\_primed\_1dpi.pdf", plot = GO\_terms\_BABA\_primed\_1dpi, width=20, height=20, dpi=300)

##———SA Primed 1dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_SA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = SA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_SA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_SA\_primed\_1dpi\_GO\_results <- GenTable(BP\_SA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_SA\_primed\_1dpi\_GO\_results  
  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_SA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = SA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_SA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_SA\_primed\_1dpi\_GO\_results <- GenTable(MF\_SA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_SA\_primed\_1dpi\_GO\_results  
  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_SA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = SA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_SA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_SA\_primed\_1dpi\_GO\_results <- GenTable(CC\_SA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_SA\_primed\_1dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_SA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(BP\_SA\_primed\_1dpi\_GO\_results$fisher)  
class(BP\_SA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_SA\_primed\_1dpi\_GO\_results\_top\_25 <- BP\_SA\_primed\_1dpi\_GO\_results[1:25,]   
BP\_SA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_SA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(BP\_SA\_primed\_1dpi\_GO\_results\_top\_25, BP\_SA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_SA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_SA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = BP\_SA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,800), breaks = seq(0,800,40))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_SA\_primed\_1dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_SA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(MF\_SA\_primed\_1dpi\_GO\_results$fisher)  
class(MF\_SA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_SA\_primed\_1dpi\_GO\_results\_top\_25 <- MF\_SA\_primed\_1dpi\_GO\_results[1:25,]   
MF\_SA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_SA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(MF\_SA\_primed\_1dpi\_GO\_results\_top\_25, MF\_SA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_SA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_SA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = MF\_SA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,800), breaks = seq(0,800,40))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_SA\_primed\_1dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_SA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(CC\_SA\_primed\_1dpi\_GO\_results$fisher)  
class(CC\_SA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_SA\_primed\_1dpi\_GO\_results\_top\_25 <- CC\_SA\_primed\_1dpi\_GO\_results[1:25,]   
CC\_SA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_SA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(CC\_SA\_primed\_1dpi\_GO\_results\_top\_25, CC\_SA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_SA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_SA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = CC\_SA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,800), breaks = seq(0,800,40))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_SA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_SA\_primed\_1dpi <- cowplot::plot\_grid(BP\_SA\_primed\_1dpi\_GO\_graph, MF\_SA\_primed\_1dpi\_GO\_graph, CC\_SA\_primed\_1dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_SA\_primed\_1dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_SA\_primed\_1dpi.pdf", plot = GO\_terms\_SA\_primed\_1dpi, width=20, height=20, dpi=300)

##———JA Primed 1dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_JA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = JA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_JA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_JA\_primed\_1dpi\_GO\_results <- GenTable(BP\_JA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_JA\_primed\_1dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_JA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = JA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_JA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_JA\_primed\_1dpi\_GO\_results <- GenTable(MF\_JA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_JA\_primed\_1dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_JA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = JA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_JA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_JA\_primed\_1dpi\_GO\_results <- GenTable(CC\_JA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_JA\_primed\_1dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_JA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(BP\_JA\_primed\_1dpi\_GO\_results$fisher)  
class(BP\_JA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_JA\_primed\_1dpi\_GO\_results\_top\_25 <- BP\_JA\_primed\_1dpi\_GO\_results[1:25,]   
BP\_JA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_JA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(BP\_JA\_primed\_1dpi\_GO\_results\_top\_25, BP\_JA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_JA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_JA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = BP\_JA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_JA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_JA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(MF\_JA\_primed\_1dpi\_GO\_results$fisher)  
class(MF\_JA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_JA\_primed\_1dpi\_GO\_results\_top\_25 <- MF\_JA\_primed\_1dpi\_GO\_results[1:25,]   
MF\_JA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_JA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(MF\_JA\_primed\_1dpi\_GO\_results\_top\_25, MF\_JA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_JA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_JA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = MF\_JA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_JA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_JA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(CC\_JA\_primed\_1dpi\_GO\_results$fisher)  
class(CC\_JA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_JA\_primed\_1dpi\_GO\_results\_top\_25 <- CC\_JA\_primed\_1dpi\_GO\_results[1:25,]   
CC\_JA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_JA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(CC\_JA\_primed\_1dpi\_GO\_results\_top\_25, CC\_JA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_JA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_JA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = CC\_JA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_JA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_JA\_primed\_1dpi <- cowplot::plot\_grid(BP\_JA\_primed\_1dpi\_GO\_graph, MF\_JA\_primed\_1dpi\_GO\_graph, CC\_JA\_primed\_1dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).  
## Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).  
## Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_JA\_primed\_1dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_JA\_primed\_1dpi.pdf", plot = GO\_terms\_JA\_primed\_1dpi, width=20, height=20, dpi=300)

#UP PRIMED 1DPI

#BABA  
#extract row names of primed DEGs  
names\_UP\_BABA\_primed\_all\_1dpi <- row.names(UP\_BABA\_primed\_all\_1dpi)  
  
#set up vector to store genes of interest for GO analysis  
UP\_BABA\_primed\_1dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data))) #can just use zero dpi raw gene data as row/gene names are the same for all three timepoints  
  
#set genes of interest to 1 and others to 0  
UP\_BABA\_primed\_1dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_UP\_BABA\_primed\_all\_1dpi))  
  
#add genes names to factor  
names(UP\_BABA\_primed\_1dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#SA  
#extract row names of primed DEGs  
names\_UP\_SA\_primed\_all\_1dpi <- row.names(UP\_SA\_primed\_all\_1dpi)  
  
#set up vector to store genes of interest for GO analysis  
UP\_SA\_primed\_1dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
UP\_SA\_primed\_1dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_UP\_SA\_primed\_all\_1dpi))  
  
#add genes names to factor  
names(UP\_SA\_primed\_1dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#JA  
#extract row names of primed DEGs  
names\_UP\_JA\_primed\_all\_1dpi <- row.names(UP\_JA\_primed\_all\_1dpi)  
  
#set up vector to store genes of interest for GO analysis  
UP\_JA\_primed\_1dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
UP\_JA\_primed\_1dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_UP\_JA\_primed\_all\_1dpi))  
  
#add genes names to factor  
names(UP\_JA\_primed\_1dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)

##———UP BABA Primed 1dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_UP\_BABA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = UP\_BABA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_UP\_BABA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_UP\_BABA\_primed\_1dpi\_GO\_results <- GenTable(BP\_UP\_BABA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_UP\_BABA\_primed\_1dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_UP\_BABA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = UP\_BABA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_UP\_BABA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_UP\_BABA\_primed\_1dpi\_GO\_results <- GenTable(MF\_UP\_BABA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_UP\_BABA\_primed\_1dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_UP\_BABA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = UP\_BABA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_UP\_BABA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_UP\_BABA\_primed\_1dpi\_GO\_results <- GenTable(CC\_UP\_BABA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_UP\_BABA\_primed\_1dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_UP\_BABA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(BP\_UP\_BABA\_primed\_1dpi\_GO\_results$fisher)  
class(BP\_UP\_BABA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25 <- BP\_UP\_BABA\_primed\_1dpi\_GO\_results[1:25,]   
BP\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_UP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(BP\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25, BP\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_UP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_UP\_BABA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = BP\_UP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_UP\_BABA\_primed\_1dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_UP\_BABA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(MF\_UP\_BABA\_primed\_1dpi\_GO\_results$fisher)  
class(MF\_UP\_BABA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25 <- MF\_UP\_BABA\_primed\_1dpi\_GO\_results[1:25,]   
MF\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_UP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(MF\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25, MF\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_UP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_UP\_BABA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = MF\_UP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_UP\_BABA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_UP\_BABA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(CC\_UP\_BABA\_primed\_1dpi\_GO\_results$fisher)  
class(CC\_UP\_BABA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25 <- CC\_UP\_BABA\_primed\_1dpi\_GO\_results[1:25,]   
CC\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_UP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(CC\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25, CC\_UP\_BABA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_UP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_UP\_BABA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = CC\_UP\_BABA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_UP\_BABA\_primed\_1dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_UP\_BABA\_primed\_1dpi <- cowplot::plot\_grid(BP\_UP\_BABA\_primed\_1dpi\_GO\_graph, MF\_UP\_BABA\_primed\_1dpi\_GO\_graph, CC\_UP\_BABA\_primed\_1dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_UP\_BABA\_primed\_1dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_UP\_BABA\_primed\_1dpi.pdf", plot = GO\_terms\_UP\_BABA\_primed\_1dpi, width=20, height=20, dpi=300)

##———UP SA Primed 1dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_UP\_SA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = UP\_SA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_UP\_SA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_UP\_SA\_primed\_1dpi\_GO\_results <- GenTable(BP\_UP\_SA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_UP\_SA\_primed\_1dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_UP\_SA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = UP\_SA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_UP\_SA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_UP\_SA\_primed\_1dpi\_GO\_results <- GenTable(MF\_UP\_SA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_UP\_SA\_primed\_1dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_UP\_SA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = UP\_SA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_UP\_SA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_UP\_SA\_primed\_1dpi\_GO\_results <- GenTable(CC\_UP\_SA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_UP\_SA\_primed\_1dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_UP\_SA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(BP\_UP\_SA\_primed\_1dpi\_GO\_results$fisher)  
class(BP\_UP\_SA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25 <- BP\_UP\_SA\_primed\_1dpi\_GO\_results[1:25,]   
BP\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_UP\_SA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(BP\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25, BP\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_UP\_SA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_UP\_SA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = BP\_UP\_SA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_UP\_SA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_UP\_SA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(MF\_UP\_SA\_primed\_1dpi\_GO\_results$fisher)  
class(MF\_UP\_SA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25 <- MF\_UP\_SA\_primed\_1dpi\_GO\_results[1:25,]   
MF\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_UP\_SA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(MF\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25, MF\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_UP\_SA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_UP\_SA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = MF\_UP\_SA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_UP\_SA\_primed\_1dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_UP\_SA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(CC\_UP\_SA\_primed\_1dpi\_GO\_results$fisher)  
class(CC\_UP\_SA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25 <- CC\_UP\_SA\_primed\_1dpi\_GO\_results[1:25,]   
CC\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_UP\_SA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(CC\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25, CC\_UP\_SA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_UP\_SA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_UP\_SA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = CC\_UP\_SA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_UP\_SA\_primed\_1dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_UP\_SA\_primed\_1dpi <- cowplot::plot\_grid(BP\_UP\_SA\_primed\_1dpi\_GO\_graph, MF\_UP\_SA\_primed\_1dpi\_GO\_graph, CC\_UP\_SA\_primed\_1dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_UP\_SA\_primed\_1dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_UP\_SA\_primed\_1dpi.pdf", plot = GO\_terms\_UP\_SA\_primed\_1dpi, width=20, height=20, dpi=300)

##———UP JA Primed 1dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_UP\_JA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = UP\_JA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_UP\_JA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_UP\_JA\_primed\_1dpi\_GO\_results <- GenTable(BP\_UP\_JA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_UP\_JA\_primed\_1dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_UP\_JA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = UP\_JA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_UP\_JA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_UP\_JA\_primed\_1dpi\_GO\_results <- GenTable(MF\_UP\_JA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_UP\_JA\_primed\_1dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_UP\_JA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = UP\_JA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_UP\_JA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_UP\_JA\_primed\_1dpi\_GO\_results <- GenTable(CC\_UP\_JA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_UP\_JA\_primed\_1dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_UP\_JA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(BP\_UP\_JA\_primed\_1dpi\_GO\_results$fisher)  
class(BP\_UP\_JA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25 <- BP\_UP\_JA\_primed\_1dpi\_GO\_results[1:25,]   
BP\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_UP\_JA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(BP\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25, BP\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_UP\_JA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_UP\_JA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = BP\_UP\_JA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_UP\_JA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_UP\_JA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(MF\_UP\_JA\_primed\_1dpi\_GO\_results$fisher)  
class(MF\_UP\_JA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25 <- MF\_UP\_JA\_primed\_1dpi\_GO\_results[1:25,]   
MF\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_UP\_JA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(MF\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25, MF\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_UP\_JA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_UP\_JA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = MF\_UP\_JA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_UP\_JA\_primed\_1dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_UP\_JA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(CC\_UP\_JA\_primed\_1dpi\_GO\_results$fisher)  
class(CC\_UP\_JA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25 <- CC\_UP\_JA\_primed\_1dpi\_GO\_results[1:25,]   
CC\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_UP\_JA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(CC\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25, CC\_UP\_JA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_UP\_JA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_UP\_JA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = CC\_UP\_JA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_UP\_JA\_primed\_1dpi\_GO\_graph

## Warning: Removed 3 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_UP\_JA\_primed\_1dpi <- cowplot::plot\_grid(BP\_UP\_JA\_primed\_1dpi\_GO\_graph, MF\_UP\_JA\_primed\_1dpi\_GO\_graph, CC\_UP\_JA\_primed\_1dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 3 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_UP\_JA\_primed\_1dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_UP\_JA\_primed\_1dpi.pdf", plot = GO\_terms\_UP\_JA\_primed\_1dpi, width=20, height=20, dpi=300)

#DOWN PRIMED 1DPI

#BABA  
#extract row names of primed DEGs  
names\_DOWN\_BABA\_primed\_all\_1dpi <- row.names(DOWN\_BABA\_primed\_all\_1dpi)  
  
#set up vector to store genes of interest for GO analysis  
DOWN\_BABA\_primed\_1dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data))) #can just use zero dpi raw gene data as row/gene names are the same for all three timepoints  
  
#set genes of interest to 1 and others to 0  
DOWN\_BABA\_primed\_1dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_DOWN\_BABA\_primed\_all\_1dpi))  
  
#add genes names to factor  
names(DOWN\_BABA\_primed\_1dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#SA  
#extract row names of primed DEGs  
names\_DOWN\_SA\_primed\_all\_1dpi <- row.names(DOWN\_SA\_primed\_all\_1dpi)  
  
#set up vector to store genes of interest for GO analysis  
DOWN\_SA\_primed\_1dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
DOWN\_SA\_primed\_1dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_DOWN\_SA\_primed\_all\_1dpi))  
  
#add genes names to factor  
names(DOWN\_SA\_primed\_1dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#JA  
#extract row names of primed DEGs  
names\_DOWN\_JA\_primed\_all\_1dpi <- row.names(DOWN\_JA\_primed\_all\_1dpi)  
  
#set up vector to store genes of interest for GO analysis  
DOWN\_JA\_primed\_1dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
DOWN\_JA\_primed\_1dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_DOWN\_JA\_primed\_all\_1dpi))  
  
#add genes names to factor  
names(DOWN\_JA\_primed\_1dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)

##———DOWN BABA Primed 1dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_DOWN\_BABA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = DOWN\_BABA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_DOWN\_BABA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results <- GenTable(BP\_DOWN\_BABA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_DOWN\_BABA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = DOWN\_BABA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_DOWN\_BABA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results <- GenTable(MF\_DOWN\_BABA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_DOWN\_BABA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = DOWN\_BABA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_DOWN\_BABA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results <- GenTable(CC\_DOWN\_BABA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results$fisher)  
class(BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25 <- BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results[1:25,]   
BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25, BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_DOWN\_BABA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,200), breaks = seq(0,200,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_DOWN\_BABA\_primed\_1dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results$fisher)  
class(MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25 <- MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results[1:25,]   
MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25, MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_DOWN\_BABA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,200), breaks = seq(0,200,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_DOWN\_BABA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results$fisher)  
class(CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25 <- CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results[1:25,]   
CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25, CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_DOWN\_BABA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,200), breaks = seq(0,200,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_DOWN\_BABA\_primed\_1dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_DOWN\_BABA\_primed\_1dpi <- cowplot::plot\_grid(BP\_DOWN\_BABA\_primed\_1dpi\_GO\_graph, MF\_DOWN\_BABA\_primed\_1dpi\_GO\_graph, CC\_DOWN\_BABA\_primed\_1dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_DOWN\_BABA\_primed\_1dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_DOWN\_BABA\_primed\_1dpi.pdf", plot = GO\_terms\_DOWN\_BABA\_primed\_1dpi, width=20, height=20, dpi=300)

##———DOWN SA Primed 1dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_DOWN\_SA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = DOWN\_SA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_DOWN\_SA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_DOWN\_SA\_primed\_1dpi\_GO\_results <- GenTable(BP\_DOWN\_SA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_DOWN\_SA\_primed\_1dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_DOWN\_SA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = DOWN\_SA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_DOWN\_SA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_DOWN\_SA\_primed\_1dpi\_GO\_results <- GenTable(MF\_DOWN\_SA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_DOWN\_SA\_primed\_1dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_DOWN\_SA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = DOWN\_SA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_DOWN\_SA\_primed\_1dpi\_topGOdata, algorithm = "elim", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_DOWN\_SA\_primed\_1dpi\_GO\_results <- GenTable(CC\_DOWN\_SA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_DOWN\_SA\_primed\_1dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_DOWN\_SA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(BP\_DOWN\_SA\_primed\_1dpi\_GO\_results$fisher)  
class(BP\_DOWN\_SA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25 <- BP\_DOWN\_SA\_primed\_1dpi\_GO\_results[1:25,]   
BP\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_DOWN\_SA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(BP\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25, BP\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_DOWN\_SA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_DOWN\_SA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = BP\_DOWN\_SA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_DOWN\_SA\_primed\_1dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_DOWN\_SA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(MF\_DOWN\_SA\_primed\_1dpi\_GO\_results$fisher)  
class(MF\_DOWN\_SA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25 <- MF\_DOWN\_SA\_primed\_1dpi\_GO\_results[1:25,]   
MF\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_DOWN\_SA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(MF\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25, MF\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_DOWN\_SA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_DOWN\_SA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = MF\_DOWN\_SA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_DOWN\_SA\_primed\_1dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_DOWN\_SA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(CC\_DOWN\_SA\_primed\_1dpi\_GO\_results$fisher)  
class(CC\_DOWN\_SA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25 <- CC\_DOWN\_SA\_primed\_1dpi\_GO\_results[1:25,]   
CC\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_DOWN\_SA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(CC\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25, CC\_DOWN\_SA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_DOWN\_SA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_DOWN\_SA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = CC\_DOWN\_SA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,140), breaks = seq(0,140,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_DOWN\_SA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_DOWN\_SA\_primed\_1dpi <- cowplot::plot\_grid(BP\_DOWN\_SA\_primed\_1dpi\_GO\_graph, MF\_DOWN\_SA\_primed\_1dpi\_GO\_graph, CC\_DOWN\_SA\_primed\_1dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_DOWN\_SA\_primed\_1dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_DOWN\_SA\_primed\_1dpi.pdf", plot = GO\_terms\_DOWN\_SA\_primed\_1dpi, width=20, height=20, dpi=300)

##———DOWN JA Primed 1dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_DOWN\_JA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = DOWN\_JA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_DOWN\_JA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_DOWN\_JA\_primed\_1dpi\_GO\_results <- GenTable(BP\_DOWN\_JA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_DOWN\_JA\_primed\_1dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_DOWN\_JA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = DOWN\_JA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_DOWN\_JA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_DOWN\_JA\_primed\_1dpi\_GO\_results <- GenTable(MF\_DOWN\_JA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_DOWN\_JA\_primed\_1dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_DOWN\_JA\_primed\_1dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = DOWN\_JA\_primed\_1dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_DOWN\_JA\_primed\_1dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_DOWN\_JA\_primed\_1dpi\_GO\_results <- GenTable(CC\_DOWN\_JA\_primed\_1dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_DOWN\_JA\_primed\_1dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_DOWN\_JA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(BP\_DOWN\_JA\_primed\_1dpi\_GO\_results$fisher)  
class(BP\_DOWN\_JA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25 <- BP\_DOWN\_JA\_primed\_1dpi\_GO\_results[1:25,]   
BP\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_DOWN\_JA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(BP\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25, BP\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_DOWN\_JA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_DOWN\_JA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = BP\_DOWN\_JA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,80), breaks = seq(0,80,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_DOWN\_JA\_primed\_1dpi\_GO\_graph

## Warning: Removed 3 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_DOWN\_JA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(MF\_DOWN\_JA\_primed\_1dpi\_GO\_results$fisher)  
class(MF\_DOWN\_JA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25 <- MF\_DOWN\_JA\_primed\_1dpi\_GO\_results[1:25,]   
MF\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_DOWN\_JA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(MF\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25, MF\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_DOWN\_JA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_DOWN\_JA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = MF\_DOWN\_JA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,80), breaks = seq(0,80,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_DOWN\_JA\_primed\_1dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_DOWN\_JA\_primed\_1dpi\_GO\_results$fisher <- as.numeric(CC\_DOWN\_JA\_primed\_1dpi\_GO\_results$fisher)  
class(CC\_DOWN\_JA\_primed\_1dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25 <- CC\_DOWN\_JA\_primed\_1dpi\_GO\_results[1:25,]   
CC\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_DOWN\_JA\_primed\_1dpi\_GO\_results\_plot\_data <- subset(CC\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25, CC\_DOWN\_JA\_primed\_1dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_DOWN\_JA\_primed\_1dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_DOWN\_JA\_primed\_1dpi\_GO\_graph <-   
 ggplot(data = CC\_DOWN\_JA\_primed\_1dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,80), breaks = seq(0,80,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_DOWN\_JA\_primed\_1dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_DOWN\_JA\_primed\_1dpi <- cowplot::plot\_grid(BP\_DOWN\_JA\_primed\_1dpi\_GO\_graph, MF\_DOWN\_JA\_primed\_1dpi\_GO\_graph, CC\_DOWN\_JA\_primed\_1dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 3 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_DOWN\_JA\_primed\_1dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_DOWN\_JA\_primed\_1dpi.pdf", plot = GO\_terms\_DOWN\_JA\_primed\_1dpi, width=20, height=20, dpi=300)

#PRIMED 2DPI

#BABA  
#extract row names of primed DEGs  
names\_BABA\_primed\_all\_2dpi <- row.names(BABA\_primed\_all\_2dpi)  
  
#set up vector to store genes of interest for GO analysis  
BABA\_primed\_2dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data))) #can just use zero dpi raw gene data as row/gene names are the same for all three timepoints  
  
#set genes of interest to 1 and others to 0  
BABA\_primed\_2dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_BABA\_primed\_all\_2dpi))  
  
#add genes names to factor  
names(BABA\_primed\_2dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#SA  
#extract row names of primed DEGs  
names\_SA\_primed\_all\_2dpi <- row.names(SA\_primed\_all\_2dpi)  
  
#set up vector to store genes of interest for GO analysis  
SA\_primed\_2dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
SA\_primed\_2dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_SA\_primed\_all\_2dpi))  
  
#add genes names to factor  
names(SA\_primed\_2dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#JA  
#extract row names of primed DEGs  
names\_JA\_primed\_all\_2dpi <- row.names(JA\_primed\_all\_2dpi)  
  
#set up vector to store genes of interest for GO analysis  
JA\_primed\_2dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
JA\_primed\_2dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_JA\_primed\_all\_2dpi))  
  
#add genes names to factor  
names(JA\_primed\_2dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)

##———BABA Primed 2dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_BABA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = BABA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_BABA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_BABA\_primed\_2dpi\_GO\_results <- GenTable(BP\_BABA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_BABA\_primed\_2dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_BABA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = BABA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_BABA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_BABA\_primed\_2dpi\_GO\_results <- GenTable(MF\_BABA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_BABA\_primed\_2dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_BABA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = BABA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_BABA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_BABA\_primed\_2dpi\_GO\_results <- GenTable(CC\_BABA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_BABA\_primed\_2dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_BABA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(BP\_BABA\_primed\_2dpi\_GO\_results$fisher)  
class(BP\_BABA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_BABA\_primed\_2dpi\_GO\_results\_top\_25 <- BP\_BABA\_primed\_2dpi\_GO\_results[1:25,]   
BP\_BABA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(BP\_BABA\_primed\_2dpi\_GO\_results\_top\_25, BP\_BABA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_BABA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = BP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_BABA\_primed\_2dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_BABA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(MF\_BABA\_primed\_2dpi\_GO\_results$fisher)  
class(MF\_BABA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_BABA\_primed\_2dpi\_GO\_results\_top\_25 <- MF\_BABA\_primed\_2dpi\_GO\_results[1:25,]   
MF\_BABA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_BABA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(MF\_BABA\_primed\_2dpi\_GO\_results\_top\_25, MF\_BABA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_BABA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_BABA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = MF\_BABA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_BABA\_primed\_2dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_BABA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(CC\_BABA\_primed\_2dpi\_GO\_results$fisher)  
class(CC\_BABA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_BABA\_primed\_2dpi\_GO\_results\_top\_25 <- CC\_BABA\_primed\_2dpi\_GO\_results[1:25,]   
CC\_BABA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_BABA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(CC\_BABA\_primed\_2dpi\_GO\_results\_top\_25, CC\_BABA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_BABA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_BABA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = CC\_BABA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_BABA\_primed\_2dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_BABA\_primed\_2dpi <- cowplot::plot\_grid(BP\_BABA\_primed\_2dpi\_GO\_graph, MF\_BABA\_primed\_2dpi\_GO\_graph, CC\_BABA\_primed\_2dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).  
## Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).  
## Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_BABA\_primed\_2dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_BABA\_primed\_2dpi.pdf", plot = GO\_terms\_BABA\_primed\_2dpi, width=20, height=20, dpi=300)

##———SA Primed 2dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_SA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = SA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_SA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_SA\_primed\_2dpi\_GO\_results <- GenTable(BP\_SA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_SA\_primed\_2dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_SA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = SA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_SA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_SA\_primed\_2dpi\_GO\_results <- GenTable(MF\_SA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_SA\_primed\_2dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_SA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = SA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_SA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_SA\_primed\_2dpi\_GO\_results <- GenTable(CC\_SA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_SA\_primed\_2dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_SA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(BP\_SA\_primed\_2dpi\_GO\_results$fisher)  
class(BP\_SA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_SA\_primed\_2dpi\_GO\_results\_top\_25 <- BP\_SA\_primed\_2dpi\_GO\_results[1:25,]   
BP\_SA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_SA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(BP\_SA\_primed\_2dpi\_GO\_results\_top\_25, BP\_SA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_SA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_SA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = BP\_SA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,720), breaks = seq(0,720,40))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_SA\_primed\_2dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_SA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(MF\_SA\_primed\_2dpi\_GO\_results$fisher)  
class(MF\_SA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_SA\_primed\_2dpi\_GO\_results\_top\_25 <- MF\_SA\_primed\_2dpi\_GO\_results[1:25,]   
MF\_SA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_SA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(MF\_SA\_primed\_2dpi\_GO\_results\_top\_25, MF\_SA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_SA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_SA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = MF\_SA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,720), breaks = seq(0,720,40))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_SA\_primed\_2dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_SA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(CC\_SA\_primed\_2dpi\_GO\_results$fisher)  
class(CC\_SA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_SA\_primed\_2dpi\_GO\_results\_top\_25 <- CC\_SA\_primed\_2dpi\_GO\_results[1:25,]   
CC\_SA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_SA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(CC\_SA\_primed\_2dpi\_GO\_results\_top\_25, CC\_SA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_SA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_SA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = CC\_SA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,720), breaks = seq(0,720,40))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_SA\_primed\_2dpi\_GO\_graph

#arrange plots and save  
GO\_terms\_SA\_primed\_2dpi <- cowplot::plot\_grid(BP\_SA\_primed\_2dpi\_GO\_graph, MF\_SA\_primed\_2dpi\_GO\_graph, CC\_SA\_primed\_2dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_SA\_primed\_2dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_SA\_primed\_2dpi.pdf", plot = GO\_terms\_SA\_primed\_2dpi, width=20, height=20, dpi=300)

##———JA Primed 2dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_JA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = JA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_JA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_JA\_primed\_2dpi\_GO\_results <- GenTable(BP\_JA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_JA\_primed\_2dpi\_GO\_results  
  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_JA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = JA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_JA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_JA\_primed\_2dpi\_GO\_results <- GenTable(MF\_JA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_JA\_primed\_2dpi\_GO\_results  
  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_JA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = JA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_JA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_JA\_primed\_2dpi\_GO\_results <- GenTable(CC\_JA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_JA\_primed\_2dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_JA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(BP\_JA\_primed\_2dpi\_GO\_results$fisher)  
class(BP\_JA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_JA\_primed\_2dpi\_GO\_results\_top\_25 <- BP\_JA\_primed\_2dpi\_GO\_results[1:25,]   
BP\_JA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_JA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(BP\_JA\_primed\_2dpi\_GO\_results\_top\_25, BP\_JA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_JA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_JA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = BP\_JA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_JA\_primed\_2dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_JA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(MF\_JA\_primed\_2dpi\_GO\_results$fisher)  
class(MF\_JA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_JA\_primed\_2dpi\_GO\_results\_top\_25 <- MF\_JA\_primed\_2dpi\_GO\_results[1:25,]   
MF\_JA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_JA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(MF\_JA\_primed\_2dpi\_GO\_results\_top\_25, MF\_JA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_JA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_JA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = MF\_JA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_JA\_primed\_2dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_JA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(CC\_JA\_primed\_2dpi\_GO\_results$fisher)  
class(CC\_JA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_JA\_primed\_2dpi\_GO\_results\_top\_25 <- CC\_JA\_primed\_2dpi\_GO\_results[1:25,]   
CC\_JA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_JA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(CC\_JA\_primed\_2dpi\_GO\_results\_top\_25, CC\_JA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_JA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_JA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = CC\_JA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_JA\_primed\_2dpi\_GO\_graph

## Warning: Removed 3 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_JA\_primed\_2dpi <- cowplot::plot\_grid(BP\_JA\_primed\_2dpi\_GO\_graph, MF\_JA\_primed\_2dpi\_GO\_graph, CC\_JA\_primed\_2dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 3 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_JA\_primed\_2dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_JA\_primed\_2dpi.pdf", plot = GO\_terms\_JA\_primed\_2dpi, width=20, height=20, dpi=300)

#UP PRIMED 2DPI

#BABA  
#extract row names of primed DEGs  
names\_UP\_BABA\_primed\_all\_2dpi <- row.names(UP\_BABA\_primed\_all\_2dpi)  
  
#set up vector to store genes of interest for GO analysis  
UP\_BABA\_primed\_2dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data))) #can just use zero dpi raw gene data as row/gene names are the same for all three timepoints  
  
#set genes of interest to 1 and others to 0  
UP\_BABA\_primed\_2dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_UP\_BABA\_primed\_all\_2dpi))  
  
#add genes names to factor  
names(UP\_BABA\_primed\_2dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#SA  
#extract row names of primed DEGs  
names\_UP\_SA\_primed\_all\_2dpi <- row.names(UP\_SA\_primed\_all\_2dpi)  
  
#set up vector to store genes of interest for GO analysis  
UP\_SA\_primed\_2dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
UP\_SA\_primed\_2dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_UP\_SA\_primed\_all\_2dpi))  
  
#add genes names to factor  
names(UP\_SA\_primed\_2dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#JA  
#extract row names of primed DEGs  
names\_UP\_JA\_primed\_all\_2dpi <- row.names(UP\_JA\_primed\_all\_2dpi)  
  
#set up vector to store genes of interest for GO analysis  
UP\_JA\_primed\_2dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
UP\_JA\_primed\_2dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_UP\_JA\_primed\_all\_2dpi))  
  
#add genes names to factor  
names(UP\_JA\_primed\_2dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)

##———UP BABA Primed 2dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_UP\_BABA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = UP\_BABA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_UP\_BABA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_UP\_BABA\_primed\_2dpi\_GO\_results <- GenTable(BP\_UP\_BABA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_UP\_BABA\_primed\_2dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_UP\_BABA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = UP\_BABA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_UP\_BABA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_UP\_BABA\_primed\_2dpi\_GO\_results <- GenTable(MF\_UP\_BABA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_UP\_BABA\_primed\_2dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_UP\_BABA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = UP\_BABA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_UP\_BABA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_UP\_BABA\_primed\_2dpi\_GO\_results <- GenTable(CC\_UP\_BABA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_UP\_BABA\_primed\_2dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_UP\_BABA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(BP\_UP\_BABA\_primed\_2dpi\_GO\_results$fisher)  
class(BP\_UP\_BABA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25 <- BP\_UP\_BABA\_primed\_2dpi\_GO\_results[1:25,]   
BP\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_UP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(BP\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25, BP\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_UP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_UP\_BABA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = BP\_UP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_UP\_BABA\_primed\_2dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_UP\_BABA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(MF\_UP\_BABA\_primed\_2dpi\_GO\_results$fisher)  
class(MF\_UP\_BABA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25 <- MF\_UP\_BABA\_primed\_2dpi\_GO\_results[1:25,]   
MF\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_UP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(MF\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25, MF\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_UP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_UP\_BABA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = MF\_UP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_UP\_BABA\_primed\_2dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_UP\_BABA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(CC\_UP\_BABA\_primed\_2dpi\_GO\_results$fisher)  
class(CC\_UP\_BABA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25 <- CC\_UP\_BABA\_primed\_2dpi\_GO\_results[1:25,]   
CC\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_UP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(CC\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25, CC\_UP\_BABA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_UP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_UP\_BABA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = CC\_UP\_BABA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_UP\_BABA\_primed\_2dpi\_GO\_graph

#arrange plots and save  
GO\_terms\_UP\_BABA\_primed\_2dpi <- cowplot::plot\_grid(BP\_UP\_BABA\_primed\_2dpi\_GO\_graph, MF\_UP\_BABA\_primed\_2dpi\_GO\_graph, CC\_UP\_BABA\_primed\_2dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")  
GO\_terms\_UP\_BABA\_primed\_2dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_UP\_BABA\_primed\_2dpi.pdf", plot = GO\_terms\_UP\_BABA\_primed\_2dpi, width=20, height=20, dpi=300)

##———UP SA Primed 2dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_UP\_SA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = UP\_SA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_UP\_SA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_UP\_SA\_primed\_2dpi\_GO\_results <- GenTable(BP\_UP\_SA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_UP\_SA\_primed\_2dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_UP\_SA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = UP\_SA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_UP\_SA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_UP\_SA\_primed\_2dpi\_GO\_results <- GenTable(MF\_UP\_SA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_UP\_SA\_primed\_2dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_UP\_SA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = UP\_SA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_UP\_SA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_UP\_SA\_primed\_2dpi\_GO\_results <- GenTable(CC\_UP\_SA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_UP\_SA\_primed\_2dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_UP\_SA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(BP\_UP\_SA\_primed\_2dpi\_GO\_results$fisher)  
class(BP\_UP\_SA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25 <- BP\_UP\_SA\_primed\_2dpi\_GO\_results[1:25,]   
BP\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_UP\_SA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(BP\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25, BP\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_UP\_SA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_UP\_SA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = BP\_UP\_SA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_UP\_SA\_primed\_2dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_UP\_SA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(MF\_UP\_SA\_primed\_2dpi\_GO\_results$fisher)  
class(MF\_UP\_SA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25 <- MF\_UP\_SA\_primed\_2dpi\_GO\_results[1:25,]   
MF\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_UP\_SA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(MF\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25, MF\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_UP\_SA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_UP\_SA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = MF\_UP\_SA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_UP\_SA\_primed\_2dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_UP\_SA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(CC\_UP\_SA\_primed\_2dpi\_GO\_results$fisher)  
class(CC\_UP\_SA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25 <- CC\_UP\_SA\_primed\_2dpi\_GO\_results[1:25,]   
CC\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_UP\_SA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(CC\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25, CC\_UP\_SA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_UP\_SA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_UP\_SA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = CC\_UP\_SA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_UP\_SA\_primed\_2dpi\_GO\_graph

#arrange plots and save  
GO\_terms\_UP\_SA\_primed\_2dpi <- cowplot::plot\_grid(BP\_UP\_SA\_primed\_2dpi\_GO\_graph, MF\_UP\_SA\_primed\_2dpi\_GO\_graph, CC\_UP\_SA\_primed\_2dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_UP\_SA\_primed\_2dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_UP\_SA\_primed\_2dpi.pdf", plot = GO\_terms\_UP\_SA\_primed\_2dpi, width=20, height=20, dpi=300)

##———UP JA Primed 2dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_UP\_JA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = UP\_JA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_UP\_JA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_UP\_JA\_primed\_2dpi\_GO\_results <- GenTable(BP\_UP\_JA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_UP\_JA\_primed\_2dpi\_GO\_results  
  
BP\_UP\_JA\_primed\_2dpi\_GO\_results$fisher <- gsub("<", "", BP\_UP\_JA\_primed\_2dpi\_GO\_results$fisher) #remove < for top p-value as wont show in graph otherwise  
  
BP\_UP\_JA\_primed\_2dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_UP\_JA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = UP\_JA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_UP\_JA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_UP\_JA\_primed\_2dpi\_GO\_results <- GenTable(MF\_UP\_JA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_UP\_JA\_primed\_2dpi\_GO\_results  
  
MF\_UP\_JA\_primed\_2dpi\_GO\_results$fisher <- gsub("<", "", MF\_UP\_JA\_primed\_2dpi\_GO\_results$fisher) #remove < for top p-value as wont show in graph otherwise  
  
MF\_UP\_JA\_primed\_2dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_UP\_JA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = UP\_JA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_UP\_JA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_UP\_JA\_primed\_2dpi\_GO\_results <- GenTable(CC\_UP\_JA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_UP\_JA\_primed\_2dpi\_GO\_results  
  
CC\_UP\_JA\_primed\_2dpi\_GO\_results$fisher <- gsub("<", "", CC\_UP\_JA\_primed\_2dpi\_GO\_results$fisher) #remove < for top p-value as wont show in graph otherwise  
  
CC\_UP\_JA\_primed\_2dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_UP\_JA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(BP\_UP\_JA\_primed\_2dpi\_GO\_results$fisher)  
class(BP\_UP\_JA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25 <- BP\_UP\_JA\_primed\_2dpi\_GO\_results[1:25,]   
BP\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_UP\_JA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(BP\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25, BP\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_UP\_JA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_UP\_JA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = BP\_UP\_JA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_UP\_JA\_primed\_2dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_UP\_JA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(MF\_UP\_JA\_primed\_2dpi\_GO\_results$fisher)  
class(MF\_UP\_JA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25 <- MF\_UP\_JA\_primed\_2dpi\_GO\_results[1:25,]   
MF\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_UP\_JA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(MF\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25, MF\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_UP\_JA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_UP\_JA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = MF\_UP\_JA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_UP\_JA\_primed\_2dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_UP\_JA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(CC\_UP\_JA\_primed\_2dpi\_GO\_results$fisher)  
class(CC\_UP\_JA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25 <- CC\_UP\_JA\_primed\_2dpi\_GO\_results[1:25,]   
CC\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_UP\_JA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(CC\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25, CC\_UP\_JA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_UP\_JA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_UP\_JA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = CC\_UP\_JA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_UP\_JA\_primed\_2dpi\_GO\_graph

#arrange plots and save  
GO\_terms\_UP\_JA\_primed\_2dpi <- cowplot::plot\_grid(BP\_UP\_JA\_primed\_2dpi\_GO\_graph, MF\_UP\_JA\_primed\_2dpi\_GO\_graph, CC\_UP\_JA\_primed\_2dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")  
GO\_terms\_UP\_JA\_primed\_2dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_UP\_JA\_primed\_2dpi.pdf", plot = GO\_terms\_UP\_JA\_primed\_2dpi, width=20, height=20, dpi=300)

#DOWN PRIMED 2DPI

#BABA  
#extract row names of primed DEGs  
names\_DOWN\_BABA\_primed\_all\_2dpi <- row.names(DOWN\_BABA\_primed\_all\_2dpi)  
  
#set up vector to store genes of interest for GO analysis  
DOWN\_BABA\_primed\_2dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data))) #can just use zero dpi raw gene data as row/gene names are the same for all three timepoints  
  
#set genes of interest to 1 and others to 0  
DOWN\_BABA\_primed\_2dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_DOWN\_BABA\_primed\_all\_2dpi))  
  
#add genes names to factor  
names(DOWN\_BABA\_primed\_2dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#SA  
#extract row names of primed DEGs  
names\_DOWN\_SA\_primed\_all\_2dpi <- row.names(DOWN\_SA\_primed\_all\_2dpi)  
  
#set up vector to store genes of interest for GO analysis  
DOWN\_SA\_primed\_2dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
DOWN\_SA\_primed\_2dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_DOWN\_SA\_primed\_all\_2dpi))  
  
#add genes names to factor  
names(DOWN\_SA\_primed\_2dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)  
  
#JA  
#extract row names of primed DEGs  
names\_DOWN\_JA\_primed\_all\_2dpi <- row.names(DOWN\_JA\_primed\_all\_2dpi)  
  
#set up vector to store genes of interest for GO analysis  
DOWN\_JA\_primed\_2dpi\_GO\_genes <- vector(length = length(row.names(zero\_dpi\_raw\_gene\_data)))  
  
#set genes of interest to 1 and others to 0  
DOWN\_JA\_primed\_2dpi\_GO\_genes <- factor(as.integer(row.names(zero\_dpi\_raw\_gene\_data) %in% names\_DOWN\_JA\_primed\_all\_2dpi))  
  
#add genes names to factor  
names(DOWN\_JA\_primed\_2dpi\_GO\_genes) <- row.names(zero\_dpi\_raw\_gene\_data)

##———DOWN BABA Primed 2dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_DOWN\_BABA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = DOWN\_BABA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_DOWN\_BABA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results <- GenTable(BP\_DOWN\_BABA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_DOWN\_BABA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = DOWN\_BABA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_DOWN\_BABA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results <- GenTable(MF\_DOWN\_BABA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_DOWN\_BABA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = DOWN\_BABA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_DOWN\_BABA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results <- GenTable(CC\_DOWN\_BABA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results$fisher)  
class(BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25 <- BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results[1:25,]   
BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25, BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_DOWN\_BABA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_DOWN\_BABA\_primed\_2dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results$fisher)  
class(MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25 <- MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results[1:25,]   
MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25, MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_DOWN\_BABA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_DOWN\_BABA\_primed\_2dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results$fisher)  
class(CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25 <- CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results[1:25,]   
CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25, CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_DOWN\_BABA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_DOWN\_BABA\_primed\_2dpi\_GO\_graph

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_DOWN\_BABA\_primed\_2dpi <- cowplot::plot\_grid(BP\_DOWN\_BABA\_primed\_2dpi\_GO\_graph, MF\_DOWN\_BABA\_primed\_2dpi\_GO\_graph, CC\_DOWN\_BABA\_primed\_2dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_DOWN\_BABA\_primed\_2dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_DOWN\_BABA\_primed\_2dpi.pdf", plot = GO\_terms\_DOWN\_BABA\_primed\_2dpi, width=20, height=20, dpi=300)

##———DOWN SA Primed 2dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_DOWN\_SA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = DOWN\_SA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_DOWN\_SA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_DOWN\_SA\_primed\_2dpi\_GO\_results <- GenTable(BP\_DOWN\_SA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_DOWN\_SA\_primed\_2dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_DOWN\_SA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = DOWN\_SA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_DOWN\_SA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_DOWN\_SA\_primed\_2dpi\_GO\_results <- GenTable(MF\_DOWN\_SA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_DOWN\_SA\_primed\_2dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_DOWN\_SA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = DOWN\_SA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_DOWN\_SA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_DOWN\_SA\_primed\_2dpi\_GO\_results <- GenTable(CC\_DOWN\_SA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_DOWN\_SA\_primed\_2dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_DOWN\_SA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(BP\_DOWN\_SA\_primed\_2dpi\_GO\_results$fisher)  
class(BP\_DOWN\_SA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25 <- BP\_DOWN\_SA\_primed\_2dpi\_GO\_results[1:25,]   
BP\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_DOWN\_SA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(BP\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25, BP\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_DOWN\_SA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_DOWN\_SA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = BP\_DOWN\_SA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_DOWN\_SA\_primed\_2dpi\_GO\_graph

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_DOWN\_SA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(MF\_DOWN\_SA\_primed\_2dpi\_GO\_results$fisher)  
class(MF\_DOWN\_SA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25 <- MF\_DOWN\_SA\_primed\_2dpi\_GO\_results[1:25,]   
MF\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_DOWN\_SA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(MF\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25, MF\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_DOWN\_SA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_DOWN\_SA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = MF\_DOWN\_SA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_DOWN\_SA\_primed\_2dpi\_GO\_graph

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_DOWN\_SA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(CC\_DOWN\_SA\_primed\_2dpi\_GO\_results$fisher)  
class(CC\_DOWN\_SA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25 <- CC\_DOWN\_SA\_primed\_2dpi\_GO\_results[1:25,]   
CC\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_DOWN\_SA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(CC\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25, CC\_DOWN\_SA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_DOWN\_SA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_DOWN\_SA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = CC\_DOWN\_SA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,100), breaks = seq(0,100,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_DOWN\_SA\_primed\_2dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_DOWN\_SA\_primed\_2dpi <- cowplot::plot\_grid(BP\_DOWN\_SA\_primed\_2dpi\_GO\_graph, MF\_DOWN\_SA\_primed\_2dpi\_GO\_graph, CC\_DOWN\_SA\_primed\_2dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_DOWN\_SA\_primed\_2dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_DOWN\_SA\_primed\_2dpi.pdf", plot = GO\_terms\_DOWN\_SA\_primed\_2dpi, width=20, height=20, dpi=300)

##———DOWN JA Primed 2dpi

#BP (biological process)  
  
#create topGOdata object to be used for enrichment analysis  
BP\_DOWN\_JA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "BP", allGenes = DOWN\_JA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(BP\_DOWN\_JA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
BP\_DOWN\_JA\_primed\_2dpi\_GO\_results <- GenTable(BP\_DOWN\_JA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
BP\_DOWN\_JA\_primed\_2dpi\_GO\_results  
  
#MF (molecular function)  
  
#create topGOdata object to be used for enrichment analysis  
MF\_DOWN\_JA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "MF", allGenes = DOWN\_JA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(MF\_DOWN\_JA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
MF\_DOWN\_JA\_primed\_2dpi\_GO\_results <- GenTable(MF\_DOWN\_JA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
MF\_DOWN\_JA\_primed\_2dpi\_GO\_results  
  
#CC (cellular compartment)  
  
#create topGOdata object to be used for enrichment analysis  
CC\_DOWN\_JA\_primed\_2dpi\_topGOdata <- new("topGOdata", ontology = "CC", allGenes = DOWN\_JA\_primed\_2dpi\_GO\_genes, annot = annFUN.gene2GO, gene2GO = GO\_database)  
  
#perform enrichment analysis  
fisher\_test <- runTest(CC\_DOWN\_JA\_primed\_2dpi\_topGOdata, algorithm = "weight01", statistic = "fisher") #double check which algorithm and test stat to use??  
fisher\_test  
  
#visualise enrichment analysis results (ordered by fisher p-value)  
CC\_DOWN\_JA\_primed\_2dpi\_GO\_results <- GenTable(CC\_DOWN\_JA\_primed\_2dpi\_topGOdata, fisher = fisher\_test,   
 orderBy = "fisher", topNodes = length(score(fisher\_test)), numChar=1000)  
CC\_DOWN\_JA\_primed\_2dpi\_GO\_results

###====graphs

#BP (biological process)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
BP\_DOWN\_JA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(BP\_DOWN\_JA\_primed\_2dpi\_GO\_results$fisher)  
class(BP\_DOWN\_JA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
BP\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25 <- BP\_DOWN\_JA\_primed\_2dpi\_GO\_results[1:25,]   
BP\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
BP\_DOWN\_JA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(BP\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25, BP\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
BP\_DOWN\_JA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
BP\_DOWN\_JA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = BP\_DOWN\_JA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("BP GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,120), breaks = seq(0,120,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
BP\_DOWN\_JA\_primed\_2dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#MF (molecular function)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
MF\_DOWN\_JA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(MF\_DOWN\_JA\_primed\_2dpi\_GO\_results$fisher)  
class(MF\_DOWN\_JA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
MF\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25 <- MF\_DOWN\_JA\_primed\_2dpi\_GO\_results[1:25,]   
MF\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
MF\_DOWN\_JA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(MF\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25, MF\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
MF\_DOWN\_JA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
MF\_DOWN\_JA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = MF\_DOWN\_JA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("MF GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,120), breaks = seq(0,120,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
MF\_DOWN\_JA\_primed\_2dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#CC (cellular compartment)  
  
#convert fisher p-value to numeric (from character) to be used as continous scale and allow subsetting  
CC\_DOWN\_JA\_primed\_2dpi\_GO\_results$fisher <- as.numeric(CC\_DOWN\_JA\_primed\_2dpi\_GO\_results$fisher)  
class(CC\_DOWN\_JA\_primed\_2dpi\_GO\_results$fisher)  
  
#select top 25 most sig GO terms   
CC\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25 <- CC\_DOWN\_JA\_primed\_2dpi\_GO\_results[1:25,]   
CC\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25  
  
#select only those 25 most sig GO terms with fisher p <= 0.05  
CC\_DOWN\_JA\_primed\_2dpi\_GO\_results\_plot\_data <- subset(CC\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25, CC\_DOWN\_JA\_primed\_2dpi\_GO\_results\_top\_25$fisher <= 0.05)  
CC\_DOWN\_JA\_primed\_2dpi\_GO\_results\_plot\_data  
  
#plot graph with bars ordered by fisher value  
CC\_DOWN\_JA\_primed\_2dpi\_GO\_graph <-   
 ggplot(data = CC\_DOWN\_JA\_primed\_2dpi\_GO\_results\_plot\_data, mapping = aes(x = fct\_reorder(Term, fisher), y = Significant, fill = fisher))+  
 geom\_col()+  
 xlab("CC GO Term")+  
 ylab("Number of Significant Genes")+  
 guides(fill=guide\_colorbar(title = "Fisher P-value"))+  
 scale\_y\_continuous(limits = c(0,120), breaks = seq(0,120,20))+  
 coord\_flip()+  
 scale\_fill\_viridis()  
  
CC\_DOWN\_JA\_primed\_2dpi\_GO\_graph

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

#arrange plots and save  
GO\_terms\_DOWN\_JA\_primed\_2dpi <- cowplot::plot\_grid(BP\_DOWN\_JA\_primed\_2dpi\_GO\_graph, MF\_DOWN\_JA\_primed\_2dpi\_GO\_graph, CC\_DOWN\_JA\_primed\_2dpi\_GO\_graph, ncol = 1, align = "hv", axis = "rblt")

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).  
## Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).  
## Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_col()`).

GO\_terms\_DOWN\_JA\_primed\_2dpi

ggsave(file="GO\_analysis/R\_topGO/GO\_terms\_DOWN\_JA\_primed\_2dpi.pdf", plot = GO\_terms\_DOWN\_JA\_primed\_2dpi, width=20, height=20, dpi=300)

#SUMMARY TABLES

##——–All primed GOs 1dpi

#In each case have selected terms with at least 10 sig hits and then selected top 10 most significant terms for brevity  
  
#BABA all GOs  
Summary\_BABA\_all\_1dpi <-   
 rbind(subset(BP\_BABA\_primed\_1dpi\_GO\_results, BP\_BABA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_BABA\_primed\_1dpi\_GO\_results, MF\_BABA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_BABA\_primed\_1dpi\_GO\_results, CC\_BABA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_BABA\_all\_1dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_BABA\_all\_1dpi) <- c("BABA.Term", "BABA.Significant", "BABA.fisher", "BABA.ontology") #rename cols to make final table clearer  
  
#SA all GOs  
  
Summary\_SA\_all\_1dpi <-   
 rbind(subset(BP\_SA\_primed\_1dpi\_GO\_results, BP\_SA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_SA\_primed\_1dpi\_GO\_results, MF\_SA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_SA\_primed\_1dpi\_GO\_results, CC\_SA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_SA\_all\_1dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_SA\_all\_1dpi) <- c("SA.Term", "SA.Significant", "SA.fisher", "SA.ontology") #rename cols to make final table clearer  
  
#JA all GOs  
  
Summary\_JA\_all\_1dpi <-   
 rbind(subset(BP\_JA\_primed\_1dpi\_GO\_results, BP\_JA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_JA\_primed\_1dpi\_GO\_results, MF\_JA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_JA\_primed\_1dpi\_GO\_results, CC\_JA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_JA\_all\_1dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_JA\_all\_1dpi) <- c("JA.Term", "JA.Significant", "JA.fisher", "JA.ontology") #rename cols to make final table clearer  
  
#Merge tables for all treatments  
Summary\_all\_1dpi <- as.data.table(append(Summary\_BABA\_all\_1dpi, Summary\_SA\_all\_1dpi))  
  
Summary\_all\_1dpi <- as.data.table(append(Summary\_all\_1dpi, Summary\_JA\_all\_1dpi))  
  
#Save final table  
WriteXLS(Summary\_all\_1dpi, "GO\_analysis/R\_topGO/Summary\_all\_primed\_GOs\_1dpi.xls")

##——–Upregulated primed GOs 1dpi

#In each case have selected terms with at least 10 sig hits and then selected top 10 most significant terms for brevity  
  
#BABA UP GOs  
Summary\_BABA\_UP\_1dpi <-   
 rbind(subset(BP\_UP\_BABA\_primed\_1dpi\_GO\_results, BP\_UP\_BABA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_UP\_BABA\_primed\_1dpi\_GO\_results, MF\_UP\_BABA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_UP\_BABA\_primed\_1dpi\_GO\_results, CC\_UP\_BABA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_BABA\_UP\_1dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_BABA\_UP\_1dpi) <- c("BABA.Term", "BABA.Significant", "BABA.fisher", "BABA.ontology") #rename cols to make final table clearer  
  
#SA UP GOs  
  
Summary\_SA\_UP\_1dpi <-   
 rbind(subset(BP\_UP\_SA\_primed\_1dpi\_GO\_results, BP\_UP\_SA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_UP\_SA\_primed\_1dpi\_GO\_results, MF\_UP\_SA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_UP\_SA\_primed\_1dpi\_GO\_results, CC\_UP\_SA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_SA\_UP\_1dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_SA\_UP\_1dpi) <- c("SA.Term", "SA.Significant", "SA.fisher", "SA.ontology") #rename cols to make final table clearer  
  
#JA UP GOs  
  
Summary\_JA\_UP\_1dpi <-   
 rbind(subset(BP\_UP\_JA\_primed\_1dpi\_GO\_results, BP\_UP\_JA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_UP\_JA\_primed\_1dpi\_GO\_results, MF\_UP\_JA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_UP\_JA\_primed\_1dpi\_GO\_results, CC\_UP\_JA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_JA\_UP\_1dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_JA\_UP\_1dpi) <- c("JA.Term", "JA.Significant", "JA.fisher", "JA.ontology") #rename cols to make final table clearer  
  
#Merge tables for UP treatments  
Summary\_UP\_1dpi <- as.data.table(append(Summary\_BABA\_UP\_1dpi, Summary\_SA\_UP\_1dpi))  
  
Summary\_UP\_1dpi <- as.data.table(append(Summary\_UP\_1dpi, Summary\_JA\_UP\_1dpi))  
  
#Save final table  
WriteXLS(Summary\_UP\_1dpi, "GO\_analysis/R\_topGO/Summary\_UP\_primed\_GOs\_1dpi.xls")

##——–Downregulated primed GOs 1dpi

#In each case have selected terms with at least 10 sig hits and then selected top 10 most significant terms for brevity  
  
#BABA DOWN GOs  
Summary\_BABA\_DOWN\_1dpi <-   
 rbind(subset(BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results, BP\_DOWN\_BABA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results, MF\_DOWN\_BABA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results, CC\_DOWN\_BABA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_BABA\_DOWN\_1dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_BABA\_DOWN\_1dpi) <- c("BABA.Term", "BABA.Significant", "BABA.fisher", "BABA.ontology") #rename cols to make final table clearer  
  
#SA DOWN GOs  
  
Summary\_SA\_DOWN\_1dpi <-   
 rbind(subset(BP\_DOWN\_SA\_primed\_1dpi\_GO\_results, BP\_DOWN\_SA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_DOWN\_SA\_primed\_1dpi\_GO\_results, MF\_DOWN\_SA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_DOWN\_SA\_primed\_1dpi\_GO\_results, CC\_DOWN\_SA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_SA\_DOWN\_1dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_SA\_DOWN\_1dpi) <- c("SA.Term", "SA.Significant", "SA.fisher", "SA.ontology") #rename cols to make final table clearer  
  
#JA DOWN GOs  
  
Summary\_JA\_DOWN\_1dpi <-   
 rbind(subset(BP\_DOWN\_JA\_primed\_1dpi\_GO\_results, BP\_DOWN\_JA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_DOWN\_JA\_primed\_1dpi\_GO\_results, MF\_DOWN\_JA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_DOWN\_JA\_primed\_1dpi\_GO\_results, CC\_DOWN\_JA\_primed\_1dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_JA\_DOWN\_1dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_JA\_DOWN\_1dpi) <- c("JA.Term", "JA.Significant", "JA.fisher", "JA.ontology") #rename cols to make final table clearer  
  
#Merge tables for DOWN treatments  
Summary\_DOWN\_1dpi <- as.data.table(append(Summary\_BABA\_DOWN\_1dpi, Summary\_SA\_DOWN\_1dpi))  
  
Summary\_DOWN\_1dpi <- as.data.table(append(Summary\_DOWN\_1dpi, Summary\_JA\_DOWN\_1dpi))  
  
#Save final table  
WriteXLS(Summary\_DOWN\_1dpi, "GO\_analysis/R\_topGO/Summary\_DOWN\_primed\_GOs\_1dpi.xls")

##——–All primed GOs 2dpi

#In each case have selected terms with at least 10 sig hits and then selected top 10 most significant terms for brevity  
  
#BABA all GOs  
Summary\_BABA\_all\_2dpi <-   
 rbind(subset(BP\_BABA\_primed\_2dpi\_GO\_results, BP\_BABA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_BABA\_primed\_2dpi\_GO\_results, MF\_BABA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_BABA\_primed\_2dpi\_GO\_results, CC\_BABA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_BABA\_all\_2dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_BABA\_all\_2dpi) <- c("BABA.Term", "BABA.Significant", "BABA.fisher", "BABA.ontology") #rename cols to make final table clearer  
  
#SA all GOs  
  
Summary\_SA\_all\_2dpi <-   
 rbind(subset(BP\_SA\_primed\_2dpi\_GO\_results, BP\_SA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_SA\_primed\_2dpi\_GO\_results, MF\_SA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_SA\_primed\_2dpi\_GO\_results, CC\_SA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_SA\_all\_2dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_SA\_all\_2dpi) <- c("SA.Term", "SA.Significant", "SA.fisher", "SA.ontology") #rename cols to make final table clearer  
  
#JA all GOs  
  
Summary\_JA\_all\_2dpi <-   
 rbind(subset(BP\_JA\_primed\_2dpi\_GO\_results, BP\_JA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_JA\_primed\_2dpi\_GO\_results, MF\_JA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_JA\_primed\_2dpi\_GO\_results, CC\_JA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_JA\_all\_2dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_JA\_all\_2dpi) <- c("JA.Term", "JA.Significant", "JA.fisher", "JA.ontology") #rename cols to make final table clearer  
  
#Merge tables for all treatments  
Summary\_all\_2dpi <- as.data.table(append(Summary\_BABA\_all\_2dpi, Summary\_SA\_all\_2dpi))  
  
Summary\_all\_2dpi <- as.data.table(append(Summary\_all\_2dpi, Summary\_JA\_all\_2dpi))  
  
#Save final table  
WriteXLS(Summary\_all\_2dpi, "GO\_analysis/R\_topGO/Summary\_all\_primed\_GOs\_2dpi.xls")

##——–Upregulated primed GOs 2dpi

#In each case have selected terms with at least 10 sig hits and then selected top 10 most significant terms for brevity  
  
#BABA UP GOs  
Summary\_BABA\_UP\_2dpi <-   
 rbind(subset(BP\_UP\_BABA\_primed\_2dpi\_GO\_results, BP\_UP\_BABA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_UP\_BABA\_primed\_2dpi\_GO\_results, MF\_UP\_BABA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_UP\_BABA\_primed\_2dpi\_GO\_results, CC\_UP\_BABA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_BABA\_UP\_2dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_BABA\_UP\_2dpi) <- c("BABA.Term", "BABA.Significant", "BABA.fisher", "BABA.ontology") #rename cols to make final table clearer  
  
#SA UP GOs  
  
Summary\_SA\_UP\_2dpi <-   
 rbind(subset(BP\_UP\_SA\_primed\_2dpi\_GO\_results, BP\_UP\_SA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_UP\_SA\_primed\_2dpi\_GO\_results, MF\_UP\_SA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_UP\_SA\_primed\_2dpi\_GO\_results, CC\_UP\_SA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_SA\_UP\_2dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_SA\_UP\_2dpi) <- c("SA.Term", "SA.Significant", "SA.fisher", "SA.ontology") #rename cols to make final table clearer  
  
#JA UP GOs  
  
Summary\_JA\_UP\_2dpi <-   
 rbind(subset(BP\_UP\_JA\_primed\_2dpi\_GO\_results, BP\_UP\_JA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_UP\_JA\_primed\_2dpi\_GO\_results, MF\_UP\_JA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_UP\_JA\_primed\_2dpi\_GO\_results, CC\_UP\_JA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_JA\_UP\_2dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_JA\_UP\_2dpi) <- c("JA.Term", "JA.Significant", "JA.fisher", "JA.ontology") #rename cols to make final table clearer  
  
#Merge tables for UP treatments  
Summary\_UP\_2dpi <- as.data.table(append(Summary\_BABA\_UP\_2dpi, Summary\_SA\_UP\_2dpi))  
  
Summary\_UP\_2dpi <- as.data.table(append(Summary\_UP\_2dpi, Summary\_JA\_UP\_2dpi))  
  
#Save final table  
WriteXLS(Summary\_UP\_2dpi, "GO\_analysis/R\_topGO/Summary\_UP\_primed\_GOs\_2dpi.xls")

##——–Downregulated primed GOs 2dpi

#In each case have selected terms with at least 10 sig hits and then selected top 10 most significant terms for brevity  
  
#BABA DOWN GOs  
Summary\_BABA\_DOWN\_2dpi <-   
 rbind(subset(BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results, BP\_DOWN\_BABA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results, MF\_DOWN\_BABA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results, CC\_DOWN\_BABA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_BABA\_DOWN\_2dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_BABA\_DOWN\_2dpi) <- c("BABA.Term", "BABA.Significant", "BABA.fisher", "BABA.ontology") #rename cols to make final table clearer  
  
#SA DOWN GOs  
  
Summary\_SA\_DOWN\_2dpi <-   
 rbind(subset(BP\_DOWN\_SA\_primed\_2dpi\_GO\_results, BP\_DOWN\_SA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_DOWN\_SA\_primed\_2dpi\_GO\_results, MF\_DOWN\_SA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_DOWN\_SA\_primed\_2dpi\_GO\_results, CC\_DOWN\_SA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_SA\_DOWN\_2dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_SA\_DOWN\_2dpi) <- c("SA.Term", "SA.Significant", "SA.fisher", "SA.ontology") #rename cols to make final table clearer  
  
#JA DOWN GOs  
  
Summary\_JA\_DOWN\_2dpi <-   
 rbind(subset(BP\_DOWN\_JA\_primed\_2dpi\_GO\_results, BP\_DOWN\_JA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #BP terms  
 subset(MF\_DOWN\_JA\_primed\_2dpi\_GO\_results, MF\_DOWN\_JA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")], #MF terms  
 subset(CC\_DOWN\_JA\_primed\_2dpi\_GO\_results, CC\_DOWN\_JA\_primed\_2dpi\_GO\_results$Significant >= 10)[1:10, c("Term", "Significant", "fisher")]) #CC terms  
  
Summary\_JA\_DOWN\_2dpi$Ontology <- c(rep("BP", times = 10), rep("MF", times = 10), rep("CC", times = 10)) #add ontology type column  
  
colnames(Summary\_JA\_DOWN\_2dpi) <- c("JA.Term", "JA.Significant", "JA.fisher", "JA.ontology") #rename cols to make final table clearer  
  
#Merge tables for DOWN treatments  
Summary\_DOWN\_2dpi <- as.data.table(append(Summary\_BABA\_DOWN\_2dpi, Summary\_SA\_DOWN\_2dpi))  
  
Summary\_DOWN\_2dpi <- as.data.table(append(Summary\_DOWN\_2dpi, Summary\_JA\_DOWN\_2dpi))  
  
#Save final table  
WriteXLS(Summary\_DOWN\_2dpi, "GO\_analysis/R\_topGO/Summary\_DOWN\_primed\_GOs\_2dpi.xls")

Note the summary tables were filtered by p-value in excel when producing final tables