```
BiocManager::install("topGO")
BiocManager::install("Rgraphviz")
library(topGO)
library(Rgraphviz)

gene.go <- read.delim("Gene2Term.txt", stringsAsFactors = F) # load in our GO to gene
mapped file.
View(gene.go)

gene.go <- gene.go[which(gene.go$GO_Term != ""),]
gene2GO <- tapply(gene.go$GO_Term, gene.go$Gene_ID, function(x)x) # this maps our genes to
thee go terms in an array
View(gene2GO)

pcutoff <- 0.05 # seetting a p-value for later. Change accordingly for fishers exact test

DE <- read.delim("Zebra_Finch.txt", stringsAsFactors = F) # load our Differential
expressed Data
View(DE)

tmp <- ifelse(DE$padj < pcutoff, 1, 0) # remove genes that arent't expressed below our p-
value
geneList <- tmp

names(geneList) <- unlist(lapply(strsplit(DE$Gene, split = ".", fixed = T),
function(x)x[1]))

head(geneList)

GOdata <- new("topGOdata",ontology = "BP", allGenes = geneList, geneSelectionFun =
function(x)(x == 1),annot = annFUN.gene2GO, gene2GO = gene2GO)
##^creates a topGO object for our Fishers test

resultFisher <- runTest(GOdata, algorithm = "elim", statistic = "fisher") # running
fishers test.

tab <- GenTable(GOdata, raw.p.value = resultFisher, topNodes = length(resultFisher@score),
numChar = 120) # turning it into a table

head(tab)

par(cex = 0.3)
showSigOfNodes(GOdata, score(resultFisher), firstSigNodes = 3, useInfo = "def") # Creates
a graph of 3 most significant nodes and their parents


goID <- tab["GO.ID"] # get your significant GO terms

my_data <- gene.go[!(gene.go$GO_Term %in% goID$GO.ID),] # Maop significatn GO terms to
Gene IDS

go2Gene <- tapply(my_data$Gene_ID, my_data$GO_Term, function(x)x) # Combine them into
Array
```