Overlap B73 and O2 Modules

```r
rm(list=ls())

library(jpeg)
library(dplyr)
library(tidyr)
library(tibble)
library(stringr)
library(ggplot2)

library(foreach)
library(iterators)
library(parallel)
library(doParallel)

library(WGCNA)
# library(KEGGREST)
# library(biomaRt)

set.seed(1)

# Enable WGCNA threads to speed up calculations
enableWGCNAThreads()
```

```r
###################################################
# Output folder
###################################################
output_path <- file.path(
  "/home/ycth8/data/projects/2021_05_30_summer_WGCNA/Maize_proteomics_output/2021_06_16_overlap_B73_O2_modules"
)

if(!dir.exists(output_path)){
  dir.create(output_path, showWarnings=FALSE, recursive=TRUE)
  if(!dir.exists(output_path)){
    quit(status=1)
  }
}
```

```r
##################################################
# Read in input file
##################################################

folder_path = file.path("/home/ycth8/data/projects/2021_05_30_summer_WGCNA/Maize_proteomics_output")

selected_genotype <- "B73"
lnames = load(
  file = file.path(
    folder_path,
    paste0("2021_06_10_", selected_genotype, "_step_by_step_network_construction"),
    paste0(selected_genotype, "-networkConstruction-stepByStep.RData")
  )
)
print(lnames)

# Replace B73 realted variable names
b73Expr = datExpr

index <- match(sub("^ME", "", colnames(MEs)), moduleColors)

colnames(MEs) <- paste0("ME", moduleLabels[index])

b73MEs = orderMEs(MEs, greyName = "ME0")
b73Labels = moduleLabels
b73Colors = moduleColors
b73Tree = geneTree
```

```r
79   selected_genotype <- "O2"
80   lnames = load(
81     file = file.path(
82       folder_path,
83       paste0("2021_06_10_", selected_genotype, "_step_by_step_network_construction"),
84       paste0(selected_genotype, "-networkConstruction-stepByStep.RData")
85     )
86   )
87   print(lnames)
88
89
90   # Replace O2 realted variable names
91   o2Expr = datExpr
92
93   index <- match(sub("^ME", "", colnames(MEs)), moduleColors)
94
95   colnames(MEs) <- paste0("ME", moduleLabels[index])
96
97   o2MEs = orderMEs(MEs, greyName = "ME0")
98   o2Labels = moduleLabels
99   o2Colors = moduleColors
100  o2Tree = geneTree
101
```

```r
108   # Isolate the module labels in the order they appear in ordered module eigengenes
109   b73ModuleLabels = substring(names(b73MEs), 3)
110   o2ModuleLabels = substring(names(o2MEs), 3)
111
112
113   # Convert the numeric module labels to color labels
114   b73Modules = labels2colors(as.numeric(b73ModuleLabels))
115   o2Modules = labels2colors(as.numeric(o2ModuleLabels))
116
117   # Numbers of B73 and O2 modules
118   nB73Mods = length(b73Modules)
119   nO2Mods = length(o2Modules)
120
121   # Initialize tables of p-values and of the corresponding counts
122   # pTable = matrix(0, nrow = nB73Mods, ncol = nO2Mods)
123   CountTbl = matrix(0, nrow = nB73Mods, ncol = nO2Mods)
124
125
126   for (i in 1:nB73Mods) {
127     for (cmod in 1:nO2Mods) {
128       b73Members = (b73Colors == b73Modules[i])
129       o2Members = (o2Colors == o2Modules[cmod])
130       # pTable[i, cmod] = -log10(fisher.test(b73Members, o2Members, alternative = "greater")$p.value)
131       CountTbl[i, cmod] = sum(b73Colors == b73Modules[i] & o2Colors == o2Modules[cmod])
132     }
133   }
```

```r
140    # Marginal counts (really module sizes)
141    b73ModTotals = apply(CountTbl, 1, sum)
142    o2ModTotals = apply(CountTbl, 2, sum)
143
144
145    cat(rep("\n", 2))
146    jpeg(file = file.path(output_path, "B73_vs_O2.jpg"), height = 480*2, width = 480*2)
147    par(mfrow=c(1,1))
148    par(cex = 1.0)
149    par(mar=c(10, 12, 3, 1))
150    labeledHeatmap(Matrix = CountTbl,
151                   xLabels = paste(" ", o2Modules),
152                   yLabels = paste(" ", b73Modules),
153                   colorLabels = TRUE,
154                   xSymbols = paste0("O2 ", o2Modules, ": ", o2ModTotals),
155                   ySymbols = paste0("B73 ", b73Modules, ": ", b73ModTotals),
156                   textMatrix = CountTbl,
157                   colors = greenWhiteRed(100)[50:100],
158                   main = "Correspondence of B73 set-specific and O2 set-specific modules",
159                   cex.text = 1.0, cex.lab = 1.0, setStdMargins = FALSE)
160
161    dev.off()
```

Correspondence of B73 set-specific and O2 set-specific modules

	O2 green: 7	O2 red: 7	O2 turquoise: 23	O2 black: 5	O2 brown: 18	O2 blue: 22	O2 yellow: 15	O2 grey: 37
B73 yellow: 12	1	1	6	1	0	0	0	3
B73 brown: 13	2	0	1	1	2	1	4	2
B73 red: 8	0	0	0	1	4	1	1	1
B73 green: 11	0	1	1	0	2	3	3	1
B73 turquoise: 52	4	3	8	1	4	9	3	20
B73 black: 6	0	0	1	0	2	1	1	1
B73 blue: 15	0	0	4	1	2	0	1	7
B73 grey: 17	0	2	2	0	2	7	2	2