EPG Data analysis

Daniel Kunk

2/10/2022

# EPG Statistical Analysis Code Package.

### This code was designed to preform statistical analysis on a csv file output from the Serrea Workbook. Please see .readme file to understand how to set up the .csv file so that it can be compatible with this code.

# loading the .csv file output. This file is designed off the output from the Serrea EPG workbook.   
# assigning the csv file to a data frame called epgdata.   
epgdata = read.csv('C:/Users/danie/Desktop/NALAM LAB/EPG/EPG ANALYSIS/24\_LD\_EPG DATA/DKVERSION\_24hr\_LD\_EPGDATA .csv')  
epgdata

# Running all of the K.W tests for each parameter across all light treatments using a For Loop.   
indices <- 3:121  
kruskal\_results\_list <- vector('list', length = length(indices))  
for (i in 3:121) {  
   
 kruskal\_results\_list[[i-2]] <- tryCatch({  
 kruskal.test(  
 x = epgdata[,i],  
 g = epgdata[,2],  
 na.action = na.omit  
 )  
   
 }, error = function(e) {  
 cat('Column ', i, ' did not run.')  
 return(e)  
 }   
 )  
}  
  
#Output of all the K.W. tests into a list called kruskal results list.   
for (i in 1:119) {  
 cat('Column ', i+2, ': \n')  
print(kruskal\_results\_list[[i]])  
}  
  
#  
p\_val\_dat <- data.frame(  
 column\_num = 3:121,  
 p\_vals = rep(0, 119)  
)  
  
for (i in 1:119) {  
 if (!is.null(kruskal\_results\_list[[i]]$p.value)) {  
 p\_val\_dat$p\_vals[i] <- kruskal\_results\_list[[i]]$p.value  
 } else {  
 p\_val\_dat$p\_vals[i] <- NA  
 }  
}  
p\_val\_dat$col\_names = colnames(epgdata[3:121])  
# The code below that has been hashtagged out will save the p-value data frame to the file location of choice. This file will list all the statistically significant waveforms from the K.W. tests.   
#write.csv(p\_val\_dat,"C:/Users/danie/Desktop/NALAM LAB/EPG/EPG ANALYSIS/24\_LD\_EPG DATA/p\_values\_24hrLD\_4hr\_kruskalwallis.csv", row.names = FALSE)

### The code below shows the same process except a different datatable with only two treatments, day and night, was used.

epgdata\_dvn = read.csv('C:/Users/danie/Desktop/NALAM LAB/EPG/EPG ANALYSIS/24\_LD\_EPG DATA/DKVERSION\_DvN\_LD\_EPGDATA .csv')  
  
indices <- 3:121  
kruskal\_results\_list\_dvn <- vector('list', length = length(indices))  
for (i in 3:121) {  
   
 kruskal\_results\_list\_dvn[[i-2]] <- tryCatch({  
 kruskal.test(  
 x = epgdata\_dvn[,i],  
 g = epgdata\_dvn[,2],  
 na.action = na.omit  
 )  
   
 }, error = function(e) {  
 cat('Column ', i, ' did not run.')  
 return(e)  
 }   
 )  
}  
#Output of all the K.W. tests into a list called kruskal results list.   
for (i in 1:119) {  
 cat('Column ', i+2, ': \n')  
print(kruskal\_results\_list\_dvn[[i]])  
}  
  
#  
p\_val\_dat\_dvn <- data.frame(  
 column\_num = 3:121,  
 p\_vals = rep(0, 119)  
)  
  
for (i in 1:119) {  
 if (!is.null(kruskal\_results\_list\_dvn[[i]]$p.value)) {  
 p\_val\_dat\_dvn$p\_vals[i] <- kruskal\_results\_list\_dvn[[i]]$p.value  
 } else {  
 p\_val\_dat\_dvn$p\_vals[i] <- NA  
 }  
}  
p\_val\_dat\_dvn$col\_names = colnames(epgdata\_dvn[3:121])  
  
write.csv(p\_val\_dat\_dvn,"C:/Users/danie/Desktop/NALAM LAB/EPG/EPG ANALYSIS/24\_LD\_EPG DATA/p\_values\_DvN\_4hr\_kruskalwallis.csv", row.names = FALSE)

### Only a pairwise wilcoxon test is neccesary here because we are only comparing two groups.

#running all of the pariwise wilcoxon rank sum tests: correcting using bonferroni, but other corrections might be more useful.   
indices <- 3:121  
multcomp\_results\_list <- vector('list', length = length(indices))  
for (i in 3:121) {  
   
 multcomp\_results\_list[[i-2]] <- tryCatch({  
 pairwise.wilcox.test(  
 x = epgdata[,i],  
 g = epgdata[,2],  
 na.action = na.omit  
 )  
   
 }, error = function(e) {  
 cat('Column ', i, ' did not run.')  
 return(e)  
 }   
 )  
}