step\_check.R

z3254626

2021-07-07

Checks for step in data by finding beta values at point: d being significantly lower than d + 1 AND d + 1 being significantly higher than d + 2

OR

d being significantly lower than d + 1 AND d + 1 being not different to d + 2 AND d + 2 being significantly higher than d + 3

@param measure Diversity measure to check for step @param data one\_locus\_data @param n.loci Number of loci

@return A tibble containing: measure = name of measure step\_present = Is the step present (T/F) step\_location\_start = location of start of step range step\_location\_end = location of end of step range

@title Step check

step\_check <- function(measure, data, n.loci) {  
   
 #Column containing mean beta values for the measure being tested  
 mean\_vec <- paste0(measure, "\_mean")  
   
 #Column containing sd beta values for the measure being tested  
 sd\_vec <- paste0(measure, "\_sd")  
  
 #Run higher\_lower\_same function (t.test) over each beta diversity value  
 beta\_diff <- pmap\_chr(list(m1 = data[[mean\_vec]], m2 = lead(data[[mean\_vec]]),   
 s1 = data[[sd\_vec]], s2 = lead(data[[sd\_vec]])),  
 higher\_lower\_same,  
 n1 = n.loci, n2 = n.loci)  
  
 x <- data[c("d", mean\_vec, sd\_vec)] %>% #select just d, mean, sd of measure  
 cbind(beta\_diff) %>% #add higher\_lower\_same results  
 mutate(step\_location\_start = ifelse((lag(beta\_diff) == "less" & #Is there a step at this d  
 beta\_diff == "greater"),   
 d, #Step between d and d + 1  
 ifelse(lag(beta\_diff) == "less" & #Is there a step between this d  
 beta\_diff == "notdifferent" & # and the next d  
 lead(beta\_diff) == "greater",   
 d, #Step between d and d +2  
 NA)), #else NA  
 step\_location\_end = ifelse((lag(beta\_diff) == "less" & #Is there a step at this d  
 beta\_diff == "greater"),   
 lead(d), #Step between d and d + 1  
 ifelse(lag(beta\_diff) == "less" & #Is there a step between this d  
 beta\_diff == "notdifferent" & # and the next d  
 lead(beta\_diff) == "greater",   
 lead(d, 2), #Step between d and d +2  
 NA)))  
   
 #Replace any Inf/ -Inf values with NA  
 step\_location\_start <- ifelse(!is.null((na.omit(x$step\_location\_start))),  
 (na.omit(x$step\_location\_start)), NA)  
   
 #Replace any Inf/ -Inf values with NA  
 step\_location\_end <- ifelse(!is.null((na.omit(x$step\_location\_end))),  
 (na.omit(x$step\_location\_end)), NA)  
   
 #T/F column indictating if a step is present  
 step\_present <- ifelse(!is.na(step\_location\_start), T, F)  
   
 return(tibble(measure = measure,   
 step\_present = step\_present,   
 step\_location\_start = step\_location\_start,  
 step\_location\_end = step\_location\_end  
 ))  
}