get\_results\_table.R

z3254626

2021-07-07

Takes merged\_results and creates Table 2 results under standard conditions

get\_results\_table <- function(merged\_results){  
   
 #Get full data set of all simulations  
 data <- merged\_results  
   
 #Vector stating the order you want the measures to be displayed  
 measure\_order <- c("H0b.Jac.AvLast", "H0b.Sor.AvLast",  
 "D0b.A.AvLast", "D0b.B.AvLast",  
 "H0b.Jac.AvFirst", "H0b.Sor.AvFirst",  
 "D0b.A.AvFirst", "D0b.B.AvFirst",  
 "H1b.MI.AvLast", "H1b.ShD.AvLast",   
 "D1b.A.AvLast", "D1b.B.AvLast",   
 "H1b.MI.AvFirst", "H1b.ShD.AvFirst",   
 "D1b.A.AvFirst", "D1b.B.AvFirst",  
 "H2b.GST.AvLast", "H2b.JOST.AvLast",   
 'D2b.A.AvLast', 'D2b.B.AvLast',  
 "H2b.GST.AvFirst", "H2b.JOST.AvFirst",   
 'D2b.A.AvFirst', 'D2b.B.AvFirst',  
 "BC.AvLast")  
   
 #Filter data down to default variables  
 data <- data %>%  
 filter(Measure %in% measure\_order,  
 n.samples == 20,  
 n.loci == 1000,  
 n.pops == 10)  
   
 #Make a column for false positives  
 #When step is zero, all steps detected are false  
 #Otherwise only those not at 0.5 are false  
 data <- data %>%  
 mutate(false\_pos = ifelse(step == 0, Total.Steps, Incorrect.Steps),  
 Correct.Steps = ifelse(step == 0, NA, Correct.Steps))  
   
 #Split into individual allele proportion treatments  
 data\_0\_1 <- data %>%  
 filter(p.start == 0) %>%  
 filter(p.end == 1) %>%  
 group\_by(Measure, step) %>%  
 summarise(correct = round(sum(Correct.Steps)/(n()), digits = 1),   
 false\_pos = round(sum(false\_pos)/(n()), digits = 1)) %>%  
 pivot\_longer(names\_to = "step\_type", cols = c(correct, false\_pos)) %>%  
 pivot\_wider(names\_from = step, values\_from = value) %>%  
 rename\_with(.cols = -c(Measure, step\_type), ~ paste(.x, "0\_1", sep = "\_"))  
   
 data\_0.1\_0.9 <- data %>%  
 filter(p.start == 0.1) %>%  
 filter(p.end == 0.9) %>%  
 group\_by(Measure, step) %>%  
 summarise(correct = round(sum(Correct.Steps)/(n()), digits = 1),   
 false\_pos = round(sum(false\_pos)/(n()), digits = 1)) %>%  
 pivot\_longer(names\_to = "step\_type", cols = c(correct, false\_pos)) %>%  
 pivot\_wider(names\_from = step, values\_from = value) %>%  
 rename\_with(.cols = -c(Measure, step\_type), ~ paste(.x, "0.1\_0.9", sep = "\_"))  
   
   
 data\_0\_0.5 <- data %>%  
 filter(p.start == 0) %>%  
 filter(p.end == 0.5)%>%  
 group\_by(Measure, step) %>%  
 summarise(correct = round(sum(Correct.Steps)/(n()), digits = 1),   
 false\_pos = round(sum(false\_pos)/(n()), digits = 1)) %>%  
 pivot\_longer(names\_to = "step\_type", cols = c(correct, false\_pos)) %>%  
 pivot\_wider(names\_from = step, values\_from = value) %>%  
 rename\_with(.cols = -c(Measure, step\_type), ~ paste(.x, "0\_0.5", sep = "\_"))  
   
   
 data\_0\_0.2 <- data %>%  
 filter(p.start == 0) %>%  
 filter(p.end == 0.2) %>%  
 group\_by(Measure, step) %>%  
 summarise(correct = round(sum(Correct.Steps)/(n()), digits = 1),   
 false\_pos = round(sum(false\_pos)/(n()), digits = 1)) %>%  
 pivot\_longer(names\_to = "step\_type", cols = c(correct, false\_pos)) %>%  
 pivot\_wider(names\_from = step, values\_from = value) %>%  
 rename\_with(.cols = -c(Measure, step\_type), ~ paste(.x, "0\_0.2", sep = "\_"))  
   
   
 data\_0.3\_0.5 <- data %>%  
 filter(p.start == 0.3) %>%  
 filter(p.end == 0.5) %>%  
 group\_by(Measure, step) %>%  
 summarise(correct = round(sum(Correct.Steps)/(n()), digits = 1),   
 false\_pos = round(sum(false\_pos)/(n()), digits = 1)) %>%  
 pivot\_longer(names\_to = "step\_type", cols = c(correct, false\_pos)) %>%  
 pivot\_wider(names\_from = step, values\_from = value) %>%  
 rename\_with(.cols = -c(Measure, step\_type), ~ paste(.x, "0.3\_0.5", sep = "\_"))  
   
   
 #Put all the data back together  
 data\_merge <- data\_0\_1 %>%  
 left\_join(data\_0.1\_0.9, by = c("Measure", "step\_type"))%>%  
 left\_join(data\_0\_0.5, by = c("Measure", "step\_type"))%>%  
 left\_join(data\_0\_0.2, by = c('Measure', "step\_type"))%>%  
 left\_join(data\_0.3\_0.5, by = c("Measure", "step\_type")) %>%  
 rename\_with(.cols = -c(Measure, step\_type), ~ paste("step", .x, sep = "\_"))  
   
 #Make table in correct order  
 data\_merge <- data\_merge %>%  
 mutate(Measure = factor(Measure, levels = measure\_order)) %>%  
 arrange(Measure)   
   
 #Format NAs as "-"  
 data\_merge <- data\_merge %>%  
 mutate\_all(~replace(., is.na(.), "-"))  
   
 # Change Labels for T/F positives  
 data\_merge$step\_type[data\_merge$step\_type == "correct"] <- "True Positives"  
 data\_merge$step\_type[data\_merge$step\_type == "false\_pos"] <- "False Positives"  
   
   
 #T/F positive steps formatting  
 data\_merge[3:22] <- lapply(data\_merge[3:22], function(x) {  
   
 v <- 1:(length(x) + 4)  
   
 v[c(T, F)] <- color\_tile("transparent", "lightblue")(c(x[c(T, F)], 0, 100))  
   
 v[c(F, T)] <- color\_tile("transparent", "tomato")(c(x[c(F, T)], 0, 100))  
   
 return(v[1:length(x)])   
 })  
   
 #Make blank measure rows for easier reading  
 data\_merge$Measure <-c("H0b Jac. (AvLast)", " ", "H0b Sor. (AvLast)"," ",   
 "D0b.A (AvLast)"," ", "D0b.B (AvLast)"," ",   
 "H0b Jac. (AvFirst)"," ", "H0b Sor. (AvFirst)"," ",   
 "D0b.A (AvFirst)"," ", "D0b.B (AvFirst)"," ",   
 "H1b MI (AvLast)"," ", "H1b ShD (AvLast)"," ",   
 "D1b.A (AvLast)"," ", "D1b.B (AvLast)"," ",   
 "H1b MI (AvFirst)"," ", "H1b ShD (AvFirst)"," ",   
 "D1b.A (AvFirst)"," ", "D1b.B (AvFirst)"," ",   
 "H2b GST (AvLast)"," ", "H2b Jost-D (AvLast)"," ",   
 'D2b.A (AvLast)'," ", 'D2b.B (AvLast)'," ",   
 "H2b GST (AvFirst)"," ", "H2b Jost-D (AvFirst)"," ",   
 'D2b.A (AvFirst)'," ", 'D2b.B (AvFirst)'," ",   
 "BC (AvLast)", " ")  
   
   
 #create one big table   
 results\_table <- data\_merge %>%  
 ungroup %>%  
 kbl(col.names = c("Measure", "Type of Step", "0", "1", "5", "50", "0", "1", "5", "50",  
 "0", "1", "5", "50", "0", "1", "5", "50", "0", "1", "5", "50"),  
 escape = FALSE) %>%  
 kable\_classic(full\_width = F, html\_font = "Cambria") %>%  
 #Add stub head for each allele treatment  
 add\_header\_above(c(" " = 2,   
 "Maximal \n range: \n p = 0 - 1" = 4,   
 "Maximal range \n without fixation: \n p = 0.1 - 0.9" = 4,   
 "Halfmaximal \n range: \n p = 0 - 0.5" = 4,  
 "Narrow range \n near fixation: \n p = 0 - 0.2" = 4,  
 "Narrow range \n far from fixation: \n p = 0.3 - 0.5" = 4)  
 ) %>%  
 #Add conditional formatting  
 pack\_rows("q = 0 Measures", 1, 16)%>%  
 pack\_rows("q = 1 Measures", 17, 32) %>%  
 pack\_rows("q = 2 Measures", 33, 48)%>%  
 pack\_rows("Bray-Curtis", 49, 50)  
   
 save\_kable(results\_table, file = "./Outputs/standard\_treatment\_table.html")  
   
 return(results\_table)  
}