DataWrangling.R

leczn

2022-03-05

#Wrangle data from simulation output  
  
#Set initial conditions  
library(dplyr)  
library(tidyr)  
library(ggplot2)  
library(stringr)  
library(lognorm)  
library(glue)  
#define directory for analysis files  
expwd <- "C:/Users/Adam/Desktop/SimThesis/OU-Thesis"  
#set directory for simulation data  
datawd <- "C:/Users/Adam/Desktop/SimThesis/Pathfinder-Simulation"  
setwd(expwd) #sets working directory to analysis folder  
dsgmatrix <- read.csv("thesis\_design.csv",stringsAsFactors = FALSE) %>%   
 select(-X.1) %>%#import design matrix  
 mutate(Trial = paste("Trial",Trial.Index,sep = " "))#add readable trial col  
  
setwd(datawd) #sets working directory to data folder  
#import occupant data from sims  
occupants <- data.frame() #make an empty data frame to fill  
for (imp\_val\_occ in 1:nrow(dsgmatrix)){#begins loop to read all occupant data  
 tri\_occ <- dsgmatrix[imp\_val\_occ,"Trial.Index"]#Pull trial number  
 scen\_occ <- dsgmatrix[imp\_val\_occ,"Scenario.Num"]#Pull Scenario number   
 occ\_path <- #create file path to read occupant data from appropriate trial  
 str\_glue("Trial{tri\_occ}/T{tri\_occ}\_S{scen\_occ}\_occupants.csv")  
 temp\_occ <- #create a temporary data frame of summarized data  
 read.csv(occ\_path,stringsAsFactors = FALSE) %>%#read appropriate data  
 summarize("Max TET" = max(exit.time.s.),"Min TET" = min(exit.time.s.),  
 "Avg TET lognorm" =   
 estimateParmsLognormFromSample(exit.time.s.)[[1]],  
 "sd TET lognorm" =  
 estimateParmsLognormFromSample(exit.time.s.)[[2]],  
 "Avg TET arithmetic" = mean(exit.time.s.),  
 "sd TET arithmetic" = sd(exit.time.s.),  
 "Max Active Time" = max(active.time.s.),  
 "Min Active Time" = min(active.time.s.),  
 "Avg Distance lognorm" =   
 estimateParmsLognormFromSample(distance..m.)[[1]],  
 "sd Distance lognorm" =  
 estimateParmsLognormFromSample(distance..m.)[[2]],  
 "Avg Distance arithmetic" = mean(distance..m.),  
 "sd Distance arithmetic" = sd(distance..m.)) %>%  
 mutate("Trial.Index" = tri\_occ)#add trial number  
 occupants <- bind\_rows(temp\_occ,occupants)#bind new data with previous data  
}  
datamatrix <- #join occupant data with design matrix into single df  
 left\_join(x = dsgmatrix,y = occupants,by = "Trial.Index")  
  
#import room data  
#import initial trial  
rooms\_select <- c("time.s.","Remaining..Total.","Exited..Total.")#define cols  
exited\_select <- c("time.s.","Exited..Total.")  
remaining\_select <- c("time.s.","Remaining..Total.")  
rooms <- read.csv("Trial1/T1\_S2\_rooms.csv",stringsAsFactors = FALSE) %>%  
 select(all\_of(rooms\_select))#pick specific columns  
#create data frame of occupants exited over time  
exited <- select(rooms,all\_of(exited\_select))  
#create data frame of occupants remaining over time  
remaining <- select(rooms,all\_of(remaining\_select))#select columns  
temp\_names <- c("Time (s)","Trial 1")  
colnames(exited) <- temp\_names#rename columns  
colnames(remaining) <- temp\_names#rename columns  
for (imp\_val\_rooms in 2:nrow(dsgmatrix)) {#begins loop to read room data  
 tri\_occ <- dsgmatrix[imp\_val\_rooms,"Trial.Index"]#Pull trial number  
 scen\_occ <- dsgmatrix[imp\_val\_rooms,"Scenario.Num"]#Pull Scenario number  
 rooms\_path <- #create file path to read occupant data from appropriate trial  
 str\_glue("Trial{tri\_occ}/T{tri\_occ}\_S{scen\_occ}\_rooms.csv")  
 rooms <- read.csv(rooms\_path,stringsAsFactors = FALSE) %>%  
 #read appropriate data & select appropriate columns  
 select(all\_of(rooms\_select))  
 temp\_names[2] <- str\_glue("Trial {tri\_occ}")#modify column names  
 temp\_exits <- #create temporary df of total exited occupants  
 select(rooms,all\_of(exited\_select))  
 temp\_remaining <- #create temp df of total remaining occupants  
 select(rooms,remaining\_select)  
 colnames(temp\_exits) <- temp\_names#renames columns  
 colnames(temp\_remaining) <- temp\_names#renames columns  
 exited <- #joins data together  
 full\_join(exited,temp\_exits,by = temp\_names[1])  
 remaining <- #joins data together  
 full\_join(remaining,temp\_remaining,by = temp\_names[1])  
}  
exited <- arrange(exited,exited[,1])#sort exited df by time column  
remaining <- arrange(remaining,remaining[,1])#sort remaining df by time column  
for (rowval in 2:nrow(remaining)) {#remove NA values from 'remaining' df  
 for (colval in 1:ncol(remaining)) {  
 if (is.na(remaining[rowval,colval])) {  
 remaining[rowval,colval] <- remaining[rowval - 1,colval]  
 }}}  
for (rowval in 2:nrow(exited)) {#remove NA values from 'exited' df  
 for (colval in 1:ncol(exited)) {  
 if (is.na(exited[rowval,colval])) {  
 exited[rowval,colval] <- exited[rowval - 1, colval]  
 }}}  
#import exit data  
#create initial data frame  
doors\_select <- #define selections  
 c("time.s.","Exit1.1","Exit1.2","Exit1.3","Exit1.4","Exit1.5")  
doors <- read.csv("Trial1/T1\_S2\_doors.csv",stringsAsFactors = FALSE) %>%  
 select(all\_of(doors\_select))#create master df & select specific cols  
exit1 <- select(doors,all\_of(doors\_select[c(1,2)]))#create df for exit1  
exit2 <- select(doors,all\_of(doors\_select[c(1,3)]))#create df for exit2  
exit3 <- select(doors,all\_of(doors\_select[c(1,4)]))#create df for exit3  
exit4 <- select(doors,all\_of(doors\_select[c(1,5)]))#create df for exit4  
exit5 <- select(doors,all\_of(doors\_select[c(1,6)]))#create df for exit5  
temp\_names <- c("Time (s)","Trial 1")#define column names  
colnames(exit1) <- temp\_names#change column names for exit1  
colnames(exit2) <- temp\_names#change column names for exit2  
colnames(exit3) <- temp\_names#change column names for exit3  
colnames(exit4) <- temp\_names#change column names for exit4  
colnames(exit5) <- temp\_names#change column names for exit5  
for (imp\_val\_doors in 2:nrow(dsgmatrix)) {#bins loop toimport rest of data  
 tri\_occ <- dsgmatrix[imp\_val\_doors,"Trial.Index"]#Pull trial number  
 scen\_occ <- dsgmatrix[imp\_val\_doors,"Scenario.Num"]#Pull Scenario number  
 doors\_path <- #create file path to read occupant data from appropriate trial  
 str\_glue("Trial{tri\_occ}/T{tri\_occ}\_S{scen\_occ}\_doors.csv")  
 doors <- read.csv(doors\_path,stringsAsFactors = FALSE) %>%  
 #read appropriate data & select appropriate columns  
 select(all\_of(doors\_select))  
 temp\_names[2] <- str\_glue("Trial {tri\_occ}")#modify column names  
 temp\_exit1 <- select(doors,all\_of(doors\_select[c(1,2)]))#create temp exit 1 df  
 temp\_exit2 <- select(doors,all\_of(doors\_select[c(1,3)]))#create temp exit 2 df  
 temp\_exit3 <- select(doors,all\_of(doors\_select[c(1,4)]))#create temp exit 3 df  
 temp\_exit4 <- select(doors,all\_of(doors\_select[c(1,5)]))#create temp exit 4 df  
 temp\_exit5 <- select(doors,all\_of(doors\_select[c(1,6)]))#create temp exit 5 df  
 colnames(temp\_exit1) <- temp\_names#rename columns  
 colnames(temp\_exit2) <- temp\_names#rename columns  
 colnames(temp\_exit3) <- temp\_names#rename columns  
 colnames(temp\_exit4) <- temp\_names#rename columns  
 colnames(temp\_exit5) <- temp\_names#rename columns  
 exit1 <- full\_join(exit1,temp\_exit1,by = temp\_names[1])#joins exit1 data  
 exit2 <- full\_join(exit2,temp\_exit2,by = temp\_names[1])#joins exit2 data  
 exit3 <- full\_join(exit3,temp\_exit3,by = temp\_names[1])#joins exit3 data  
 exit4 <- full\_join(exit4,temp\_exit4,by = temp\_names[1])#joins exit4 data  
 exit5 <- full\_join(exit5,temp\_exit5,by = temp\_names[1])#joins exit5 data  
}  
exit1[is.na(exit1)] <- 0 #replace NA with zero values  
exit2[is.na(exit2)] <- 0 #replace NA with zero values  
exit3[is.na(exit3)] <- 0 #replace NA with zero values  
exit4[is.na(exit4)] <- 0 #replace NA with zero values  
exit5[is.na(exit5)] <- 0 #replace NA with zero values  
  
exit1 <- arrange(exit1,exit1[,1])#sort exit1 df by time  
exit2 <- arrange(exit2,exit2[,1])#sort exit2 df by time  
exit3 <- arrange(exit3,exit3[,1])#sort exit3 df by time  
exit4 <- arrange(exit4,exit4[,1])#sort exit4 df by time  
exit5 <- arrange(exit5,exit5[,1])#sort exit5 df by time  
  
#summarize exit data  
exit1\_cum <- arrange(exit1,exit1[,1])#create a new df for cumulative count  
exit2\_cum <- arrange(exit2,exit2[,1])#create a new df for cumulative count  
exit3\_cum <- arrange(exit3,exit3[,1])#create a new df for cumulative count  
exit4\_cum <- arrange(exit4,exit4[,1])#create a new df for cumulative count  
exit5\_cum <- arrange(exit5,exit5[,1])#create a new df for cumulative count  
  
for (cum\_exit in (2:nrow(exit1\_cum))) {#cumulative sum of all rows for exit1  
 exit1\_cum[cum\_exit,-1] <- exit1\_cum[cum\_exit,-1] + exit1\_cum[cum\_exit - 1,-1]}  
for (cum\_exit in (2:nrow(exit2\_cum))) {#cumulative sum of all rows for exit2  
 exit2\_cum[cum\_exit,-1] <- exit2\_cum[cum\_exit,-1] + exit2\_cum[cum\_exit - 1,-1]}  
for (cum\_exit in (2:nrow(exit3\_cum))) {#cumulative sum of all rows for exit3  
 exit3\_cum[cum\_exit,-1] <- exit3\_cum[cum\_exit,-1] + exit3\_cum[cum\_exit - 1,-1]}  
for (cum\_exit in (2:nrow(exit4\_cum))) {#cumulative sum of all rows for exit2  
 exit4\_cum[cum\_exit,-1] <- exit4\_cum[cum\_exit,-1] + exit4\_cum[cum\_exit - 1,-1]}  
for (cum\_exit in (2:nrow(exit5\_cum))) {#cumulative sum of all rows for exit2  
 exit5\_cum[cum\_exit,-1] <- exit5\_cum[cum\_exit,-1] + exit5\_cum[cum\_exit - 1,-1]}  
  
#define function to summarize exit data & join to occupants matrix  
sum\_exit\_data <- function (df1,df2,headername) {  
 temp\_df <- #create temporary df  
 gather(df1[,-1],key = Trial,value = temp) %>%#unpivot into single column  
 group\_by(Trial) %>%#tell formula to aggregate by trial  
 summarize("{headername}" := sum(temp)) %>%#calculate total exited per trial  
 mutate(Trial.Index = as.integer(#separate trial number from string  
 str\_sub(Trial,  
 start = 7,  
 end = str\_length(Trial)))) %>%  
 select(-Trial)  
 datamatrix <<- #join exit data to data matrix df by trial number  
 left\_join(x = df2,y = temp\_df,by = "Trial.Index")  
}  
sum\_exit\_data(df1 = exit1, df2 = datamatrix,headername = "Exit1")#execute exit1  
sum\_exit\_data(df1 = exit2, df2 = datamatrix,headername = "Exit2")#execute exit2  
sum\_exit\_data(df1 = exit3, df2 = datamatrix,headername = "Exit3")#execute exit3  
sum\_exit\_data(df1 = exit4, df2 = datamatrix,headername = "Exit4")#execute exit4  
sum\_exit\_data(df1 = exit5, df2 = datamatrix,headername = "Exit5")#execute exit5  
datamatrix <- mutate(datamatrix, TotalExit = Exit1 + Exit2 + Exit3  
 + Exit4 + Exit5)  
#clean up variables  
rm(doors,rooms,temp\_names,temp\_occ,temp\_remaining,tri\_occ,scen\_occ,doors\_path,  
 doors\_select,exited\_select,imp\_val\_rooms,imp\_val\_doors,imp\_val\_occ,  
 occ\_path,remaining\_select,temp\_exit1,temp\_exit2,temp\_exit3,temp\_exit4,  
 temp\_exit5,rooms\_select,rooms\_path,temp\_exits,cum\_exit,rowval,colval)  
  
#one-time print of data .csv files  
setwd(expwd)  
#write.csv(occupants,"occupants.csv")  
#write.csv(exit1,"exit1.csv")  
#write.csv(exit2,"exit2.csv")  
#write.csv(exit3,"exit3.csv")  
#write.csv(exit4,"exit4.csv")  
#write.csv(exit5,"exit5.csv")  
#write.csv(exited,"exited.csv")  
#write.csv(remaining,"remaining.csv")  
#write.csv(exit1\_cum,"exit1\_cum.csv")  
#write.csv(exit2\_cum,"exit2\_cum.csv")  
#write.csv(exit3\_cum,"exit3\_cum.csv")  
#write.csv(exit4\_cum,"exit4\_cum.csv")  
#write.csv(exit5\_cum,"exit5\_cum.csv")  
#write.csv(datamatrix,"datamatrix.csv")