thesis\_analysis.R

leczn

2022-03-05

#Analyze data from simulation output  
  
#Set initial conditions  
library(dplyr)  
library(tidyr)  
library(ggplot2)  
library(stringr)  
library(lognorm)  
library(RColorBrewer)  
library(ggpubr)  
library(knitr)  
#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  
#read data that was previous wrangled and printed into appropriate directory  
analysisfiles <- list.files(expwd)#list all files in current directory  
analysisfiles <- #filter for only .csv files  
 analysisfiles[str\_detect(analysisfiles,".csv")]  
for (readers in 1:length(analysisfiles)) {  
 assign(  
 str\_remove(analysisfiles[readers],".csv"),  
 read.csv(analysisfiles[readers],stringsAsFactors = FALSE))  
}  
  
#plot occupants that exited building  
vlines <- c(200,475,550) #create x intercepts for vertical lines  
exited\_plotdata <- exited %>% select(-X) %>% #final wrangling steps  
 gather(key = Trial,value = exited,-Time..s.) %>%  
 mutate(Trial = str\_replace(Trial,"Trial.","Trial ")) %>%  
 left\_join(y = select(datamatrix,Trial,Scenario.Num),by = "Trial") %>%  
 mutate(Scenario = paste("S",Scenario.Num,sep = ""))  
#exited\_labeled <- exited\_plotdata %>% filter(Trial == "Trial 32") %>%  
 #filter(Time..s. == 250) #filter for suspect trial  
exited\_plot <- #generate plot  
 ggplot(exited\_plotdata,mapping = aes(x = Time..s.,y = exited)) +  
 geom\_line(mapping = aes(colour = Scenario, group = Trial)) +  
 #geom\_label(exited\_labeled,mapping = aes(label = Trial)) +  
 labs(title = "Total Occupants Exited In Evacuation",  
 y = "Occupants Exited", x = "Time (s)") +  
 scale\_y\_continuous(limits = range(exited\_plotdata$exited)) +  
 scale\_x\_continuous(limits = range(exited\_plotdata$Time..s.),  
 n.breaks = 25) +   
 geom\_vline(xintercept = vlines,  
 color = "orange", size = 4,alpha = 0.3) +  
 annotate(geom = "text",label = str\_glue("{vlines} s"),  
 x = vlines, y = 250,angle = 90,vjust = 1)  
  
#plot occupants that remained in building  
remaining\_plotdata <- remaining %>% select(-X) %>% #final wrangling steps  
 gather(key = Trial,value = remaining,-Time..s.) %>%  
 mutate(Trial = str\_replace(Trial,"Trial.","Trial ")) %>%  
 left\_join(y = select(datamatrix,Trial,Scenario.Num),by = "Trial") %>%  
 mutate(Scenario = paste("S",Scenario.Num,sep = ""))  
#remaining\_labeled <- remaining\_plotdata %>% #filters for suspected trial  
 #filter(Trial == "Trial 32") %>% filter(Time..s. == 250)  
remaining\_plot <- #generate plot  
 ggplot(remaining\_plotdata,mapping = aes(x = Time..s.,y = remaining)) +  
 geom\_line(mapping = aes(colour = Scenario,group = Trial)) +  
 #geom\_label(data = remaining\_labeled,mapping = aes(label = Trial)) +  
 labs(title = "Total Occupants Remaining In Evacuation",  
 y = "Occupants Remaining", x = "Time (s)") +  
 scale\_y\_continuous(limits = range(remaining\_plotdata$remaining)) +  
 scale\_x\_continuous(limits = range(exited\_plotdata$Time..s.),  
 n.breaks = 25) +   
 geom\_vline(xintercept = c(200,475,550),  
 color = "orange", size = 4,alpha = 0.3) +  
 annotate(geom = "text",label = str\_glue("{vlines} s"),  
 x = vlines, y = 2250,angle = 90,vjust = 1)  
  
#plot exit data by exit  
exit1\_cum\_plot\_data <- exit1\_cum %>% select(-X) %>%#df of flattened exit1 data  
 gather(key = Trial,value = numexited,-Time..s.) %>%   
 mutate(exit = "Exit 1",Trial = str\_replace(Trial,"Trial.","Trial "))  
exit2\_cum\_plot\_data <- exit2\_cum %>% select(-X) %>%#df of flattened exit2 data  
 gather(key = Trial,value = numexited,-Time..s.) %>%   
 mutate(exit = "Exit 2",Trial = str\_replace(Trial,"Trial.","Trial "))  
exit3\_cum\_plot\_data <- exit3\_cum %>% select(-X) %>%#df of flattened exit3 data  
 gather(key = Trial,value = numexited,-Time..s.) %>%   
 mutate(exit = "Exit 3",Trial = str\_replace(Trial,"Trial.","Trial "))  
exit4\_cum\_plot\_data <- exit4\_cum %>% select(-X) %>%#df of flattened exit4 data  
 gather(key = Trial,value = numexited,-Time..s.) %>%   
 mutate(exit = "Exit 4",Trial = str\_replace(Trial,"Trial.","Trial "))  
exit5\_cum\_plot\_data <- exit5\_cum %>% select(-X) %>%#df of flattened exit5 data  
 gather(key = Trial,value = numexited,-Time..s.) %>%   
 mutate(exit = "Exit 5",Trial = str\_replace(Trial,"Trial.","Trial "))  
all\_exit\_cum\_data <- rbind(exit1\_cum\_plot\_data,exit2\_cum\_plot\_data,  
 exit3\_cum\_plot\_data,exit4\_cum\_plot\_data,  
 exit5\_cum\_plot\_data) %>% #append exit data frames  
 left\_join(y = select(datamatrix,Trial,Scenario),by = "Trial")  
#all\_exit\_labeled <- all\_exit\_cum\_data %>% #filter for suspected trial  
 #filter(Trial == "Trial 32") %>% filter(Time..s. == 250)  
all\_exit\_cum\_plot <- #generate plot  
 ggplot(all\_exit\_cum\_data, mapping = aes(x = Time..s.,y = numexited)) +  
 geom\_line(mapping = aes(colour = Scenario,group = Trial)) +  
 #geom\_label(data = all\_exit\_labeled,mapping = aes(label = Trial)) +  
 facet\_wrap(facets = ~exit) + #add facets for multiple plots  
 labs(title = "Total Occupants Exited by Exit",  
 y = "Occupants Exited",  
 x = "Time (s)")  
  
#plot proportion of occupants that exited by trial  
prop\_exits\_plot\_data <- datamatrix %>%#final data wrangling  
 select(Scenario,Trial.Index,Exit1,Exit2,Exit3,Exit4,Exit5) %>%  
 gather(key = exit,value = numexited,Exit1,Exit2,Exit3,Exit4,Exit5)  
prop\_exited\_plot <- #generate plot  
 ggplot(prop\_exits\_plot\_data) +  
 geom\_col(mapping = aes(x = Trial.Index,y = numexited,fill = exit),  
 position = "fill") +  
 facet\_wrap(facets = ~Scenario) +  
 labs(title = "Proportion of Agents choosing each Exit",  
 x = "Trial Number",y = "Proporition of Agents",fill = "Exit")  
  
#plot TET  
TET\_plot\_data <- #pick data  
 select(datamatrix,Trial,Scenario,Trial.Index,  
 Max.TET,Avg.TET.arithmetic,sd.TET.arithmetic)  
TET\_plot <- #generate plot  
 ggplot(TET\_plot\_data,mapping = aes(x = Trial.Index, colour = Scenario)) +   
 geom\_segment(mapping = aes(y = Max.TET,xend = Trial.Index+1,yend = Max.TET)) +  
 geom\_point(mapping = aes(y = Avg.TET.arithmetic,size = sd.TET.arithmetic)) +  
 labs(title = "Average and Maximum Evacuation Times",  
 x = "Trial Number",y = "Time (s)",size = "Standard Deviation")  
TET\_hist <- #plot TET histogram  
 ggplot (TET\_plot\_data,mapping = aes(x = Max.TET)) +  
 geom\_histogram(binwidth = 10) +  
 facet\_wrap(facets = ~Scenario) +  
 labs(title = "Histogram of Total Evacuation Time",  
 x = "Time (s)",y = "Count")  
TET\_box <- #plot TET boxplot  
 ggplot(TET\_plot\_data,mapping = aes(x = Scenario, y = Max.TET)) +  
 geom\_boxplot() +  
 labs(title = "Box Plot of Total Evacuation Time",  
 x = "Simulation Scenario",y = "Time (s)")  
#plot Avg Distance  
dist\_plot\_data <- #pick data  
 select(datamatrix,Trial,Scenario,Trial.Index,  
 Avg.Distance.arithmetic,sd.Distance.arithmetic)  
dist\_plot <- #generate plot  
 ggplot(dist\_plot\_data,mapping = aes(x = Trial.Index, colour = Scenario)) +  
 geom\_point(mapping = aes(y = Avg.Distance.arithmetic,  
 size = sd.Distance.arithmetic)) +  
 labs(title = "Average and Maximum Travel Distance",  
 x = "Trial Number",y = "Distance (m)",size = "Standard Deviation")  
dist\_hist <- #plot TET histogram  
 ggplot (dist\_plot\_data,mapping = aes(x = Avg.Distance.arithmetic)) +  
 geom\_histogram(binwidth = 1) +  
 facet\_wrap(facets = ~Scenario) +  
 labs(title = "Histogram of Average Travel Distance",  
 x = "Distance (m)",y = "Count")  
dist\_box <- #plot TET boxplot  
 ggplot(dist\_plot\_data,mapping = aes(x = Scenario,  
 y = Avg.Distance.arithmetic)) +  
 geom\_boxplot() +  
 labs(title = "Box Plot of Average Travel Distance",  
 x = "Simulation Scenario",y = "Distance (m)")  
  
#start ANOVA of TET  
#outsub <- "Outlier(s) Removed" #define lable for outliers removed  
anova\_df <- #make analysis df  
 datamatrix %>%  
 select(startposition,obstaclespresent,populationsize,Max.TET,  
 Avg.Distance.arithmetic,Trial.Index,Starting.Position,  
 Obstacles,Population.Size) %>%  
 mutate(startposition = as.factor(startposition),  
 obstaclespresent = as.factor(obstaclespresent),  
 populationsize = as.factor(populationsize)) #%>%  
 #filter(Trial.Index != 32) #remove Trial 32, which is an outlier  
TET\_anova <- #generate anova model  
 aov(Max.TET ~ startposition \* obstaclespresent \* populationsize,  
 data = anova\_df)  
TET\_residual\_data <- #draw out residuals & fitted values  
 data.frame(TET\_anova$residuals,TET\_anova$fitted.values) %>%  
 mutate(Trial.Index = c(1:40)) %>%   
 left\_join(select(datamatrix,Trial.Index,Scenario),by = "Trial.Index")  
TET\_residual\_plot <- #generate residual plot  
 ggplot(TET\_residual\_data,  
 mapping = aes(x = TET\_anova.fitted.values,y = TET\_anova.residuals)) +  
 geom\_point(mapping = aes(colour = Scenario)) +  
 labs(title = "Plot of Residuals of TET Values",  
 x = "Fitted TET Values (s)",  
 y = "Residuals"#,  
 #subtitle = outsub  
 )  
TET\_qq\_plot\_residuals <- ggqqplot(TET\_anova$residuals) +  
 labs(title = "Q-Q Plot of TET Residuals"#,  
 #subtitle = outsub  
 )#generate plot to test normality  
TET\_qq\_plot\_effects <- ggqqplot(TET\_anova$effects) +  
 labs(title = "Q-Q Plot of TET Effects"#,  
 #subtitle = outsub  
 )#generate plot to test normality  
TET\_population\_me\_plot <- #generate main effects plot   
 ggplot(anova\_df,mapping = aes(x = populationsize,y = Max.TET)) +  
 stat\_summary(fun = mean,geom = "line",#create connecting line  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "point",#create points for main effects  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "text",#add values labels of main effects  
 mapping = aes(label = round(..y..,2)), hjust = -0.3) +  
 scale\_x\_discrete(labels = c("Low","High")) +  
 labs(title = "Main Effects Plot for TET against Population Size",  
 x = "Population Factor Level", y = "Response: TET (s)"#,  
 #subtitle = outsub  
 )  
TET\_obstacles\_me\_plot <- #generate main effects plot   
 ggplot(anova\_df,mapping = aes(x = obstaclespresent,y = Max.TET)) +  
 stat\_summary(fun = mean,geom = "line",#create connecting line  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "point",#create points for main effects  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "text",#add values labels of main effects  
 mapping = aes(label = round(..y..,2)), hjust = -0.3) +  
 scale\_x\_discrete(labels = c("Not-Present","Present")) +  
 labs(title = "Main Effects Plot for TET against the Presence of Obstacles",  
 x = "Obstacle Factor Level", y = "Response: TET (s)"#,  
 #subtitle = outsub  
 )  
TET\_position\_me\_plot <- #generate main effects plot   
 ggplot(anova\_df,mapping = aes(x = startposition,y = Max.TET)) +  
 stat\_summary(fun = mean,geom = "line",#create connecting line  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "point",#create points for main effects  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "text", #add values labels of main effects  
 mapping = aes(label = round(..y..,2)), hjust = -0.3) +  
 scale\_x\_discrete(labels = c("Unconcentrated","Concentrated")) +  
 labs(title =   
 "Main Effects Plot for TET against the Starting Position of Agents",  
 x = "Position Factor Level", y = "Response: TET (s)"#,  
 #subtitle = outsub  
 )  
  
#start ANOVA for avg distance traveled  
#anova\_df <- #remove another outlier that exits for avg distance  
 #anova\_df %>% filter(Trial.Index != 6)  
distance\_anova <- aov(Avg.Distance.arithmetic ~ populationsize \*  
 obstaclespresent \* startposition, data = anova\_df)  
distance\_residual\_data <- #collect residuals from ANOVA output  
 data.frame(distance\_anova$residuals,distance\_anova$fitted.values) %>%  
 mutate(Trial.Index = c(1:40)) %>%  
 left\_join(select(datamatrix,Scenario,Trial.Index),by = "Trial.Index")  
distance\_residual\_plot <- #generate plot of residuals vs. fitted values  
 ggplot(data = distance\_residual\_data,  
 mapping = aes(x = distance\_anova.fitted.values,  
 y = distance\_anova.residuals)) +  
 geom\_point(mapping = aes(colour = Scenario)) +  
 labs(title = "Plot of Residuals of Average Distance Traveled",  
 x = "Fitted Distance Values (m)",y = "Residuals"#,  
 #subtitle = outsub  
 )  
distance\_qq\_plot\_residuals <- #generate qq plot of residuals  
 ggqqplot(distance\_anova$residuals) +  
 labs(title = "Q-Q Plot of Average Distance Residuals"#,  
 #subtitle = outsub  
 )  
distance\_qq\_plot\_effects <- #generate qq plot of effects  
 ggqqplot(distance\_anova$effects) +   
 labs(title = "Q-Q Plot of Average Distance Effects"#,  
 #subtitle = outsub  
 )  
distance\_population\_me\_plot <- #generate main effects plot   
 ggplot(anova\_df,mapping = aes(x = populationsize,  
 y = Avg.Distance.arithmetic)) +  
 stat\_summary(fun = mean,geom = "line",  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "point",  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "text",#add values labels of main effects  
 mapping = aes(label = round(..y..,2)), hjust = -0.3) +  
 scale\_x\_discrete(labels = c("Low","High")) +  
 labs(title =   
"Main Effects Plot for Average Distance Traveled against the Population Size",  
 x = "Population Factor Level", y = "Response: Distance Traveled (m)"#,  
 #subtitle = outsub  
 )  
distance\_obstacles\_me\_plot <- #generate main effects plot   
 ggplot(anova\_df,mapping = aes(x = obstaclespresent,  
 y = Avg.Distance.arithmetic)) +  
 stat\_summary(fun = mean,geom = "line",  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "point",  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "text",#add values labels of main effects  
 mapping = aes(label = round(..y..,2)), hjust = -0.3) +  
 scale\_x\_discrete(labels = c("Not Present","Present")) +  
 labs(title =   
"Main Effects Plot for Average Distance Traveled against the   
Presence of Obstacles",  
 x = "Obstacle Factor Level", y = "Response: Distance Traveled (m)"#,  
 #subtitle = outsub  
 )  
distance\_position\_me\_plot <- #generate main effects plot   
 ggplot(anova\_df,mapping = aes(x = startposition,  
 y = Avg.Distance.arithmetic)) +  
 stat\_summary(fun = mean,geom = "line",  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "point",  
 aes(color = "red",group = 1),show.legend = FALSE) +  
 stat\_summary(fun = mean,geom = "text",#add values labels of main effects  
 mapping = aes(label = round(..y..,2)), hjust = -0.3) +  
 scale\_x\_discrete(labels = c("Unconcentrated","Concentrated")) +  
 labs(title =   
"Main Effects Plot for Average Distance Traveled against the Starting   
Position of Agents",  
 x = "Position Factor Level", y = "Response: Distance Traveled (m)"#,  
 #subtitle = outsub  
 )  
  
#take a sample of two from each scenario group for analysis within Pathfinder  
#sample(datamatrix[datamatrix$Scenario == "S4","Trial.Index"],2)  
#S1 = 18 and 30,S2 = 29 and 24,S3 = 40 and 2,S4 = 31 and 8  
#Trial 32 will also be included since it is such a large outlier  
#take a sample of times to analyze further using Pathfinder  
#sample(c(100:225),1)#select a time from beginning of sim to analyze [183 s]  
#sample(c(250:450),1)#select a time from midle of sim to analyze [325 s]  
#sample(c(500:550),1)#select a time from end of sim for low population [504 s]  
#sample(c(600:675),1)#select a time from end of sim for high population [637 s]