

question-4-code/random_walk.R		+14 -11
@@ -1,34 +1,36 @@		
1 #install.packages("ggplot2")	1 #install.packages("ggplot2")	
2 #install.packages("gridExtra")	2 #install.packages("gridExtra")	
3	3	
4 library(ggplot2)	4 library(ggplot2)	
5 library(gridExtra)	5 library(gridExtra)	
6	6	
7 - random_walk <- function (n_steps) {	7 + random_walk <- function (n_steps, seed = 123) {	
	8 +	
	9 + set.seed(seed)	
8	10	
9 df <- data.frame(x = rep(NA, n_steps), y = rep(NA,	11 df <- data.frame(x = rep(NA, n_steps), y = rep(NA,	
n_steps), time = 1:n_steps)	n_steps), time = 1:n_steps)	
10	12	
11 - df[1,] <- c(0,0,1)	13 + df[1,] <- c(0,0,1) # Starting point of random walk	
12	14	
13 - for (i in 2:n_steps) {	15 + for (i in 2:n_steps) { # For loop starts at i = 2 and	
	runs until i = n_steps	
14	16	
15 - h <- 0.25	17 + h <- 0.25 # Constant height	
16	18	
17 - angle <- runif(1, min = 0, max = 2*pi)	19 + angle <- runif(1, min = 0, max = 2*pi) # random	
18	angle generated between 0 and 2 pi	
19 - df[i,1] <- df[i-1,1] + cos(angle)*h	21 + df[i,1] <- df[i-1,1] + cos(angle)*h # cos() changes	
20	x-direction	
21 - df[i,2] <- df[i-1,2] + sin(angle)*h	23 + df[i,2] <- df[i-1,2] + sin(angle)*h # sin() changes	
22	y direction	
23 - df[i,3] <- i	25 + df[i,3] <- i	
24	26	
25 }	27 }	
26	28	
27 - return(df)	29 + return(df)	
28	30	
29 }	31 }	
30	32	
31 - data1 <- random_walk(500)	33 + data1 <- random_walk(500, seed = 123) # 500 steps in	
32	this random walk (with seed)	
33 plot1 <- ggplot(aes(x = x, y = y), data = data1) +	34	
34	35 plot1 <- ggplot(aes(x = x, y = y), data = data1) +	
	36	
@@ -40,7 +42,7 @@		
40	42	
41 ylab("y-coordinate")	43 ylab("y-coordinate")	
42	44	
43 - data2 <- random_walk(500)	45 + data2 <- random_walk(500, seed = 123) # 500 steps in	
44	this random walk (with same seed so will be the same)	
45 plot2 <- ggplot(aes(x = x, y = y), data = data2) +	46	
46	47 plot2 <- ggplot(aes(x = x, y = y), data = data2) +	
	48	
@@ -53,3 +55,4 @@		
53 ylab("y-coordinate")	55 ylab("y-coordinate")	
54	56	
55 grid.arrange(plot1, plot2, ncol=2)	57 grid.arrange(plot1, plot2, ncol=2)	
	58 +	