

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