**S291-Homework1**

**Group Members:**

Kemal Doğu Oktay – 2502250

Sefa Bulut – 2501116

Furkan Kazancıoğlu – 2502185

Rubar Akyıldız – 2561033

Ozan Adıgüzel – 2458404

**##### QUESTION 1 #####**

**### Part a: Generate the following variables.**

**# 1. 10 observations from Uniform(-2, 2). Name it as unif.**

unif <- runif(10, min = -2, max = 2)

unif

**# 2. 10 observations from Normal(mean = 5, sd = 4). Name it as norm.**

norm <- rnorm(10, mean = 5, sd = 4)

norm

**# 3. 10 capital letters selected with replacement. Name it as LET.**

LET <- sample(LETTERS, 10, replace = TRUE)

LET

### Part b: **Create a data frame including the variables in part (a). Print**

**# the resulted data frame.**

df <- data.frame(unif, norm, LET)

df

**### Part c: Add a new column of ab = (A,A,A,A,A,B,B,B,B,B) to your data**

**#frame. Print the resulted data frame.**

df$ab <- c('A', 'A', 'A', 'A', 'A', 'B', 'B', 'B', 'B', 'B')

df

**### Part d: Create a new dataframe including only positive values for the**

**#observations generated from Uniform distribution (unif column). You should**

**#include the other variables also. Then, calculate the standard deviation of**

**#observations under unif column.**

df2 <- df[df$unif > 0,]

df2

std\_dev <- sd(df2$unif)

std\_dev

df3 <- rbind(df2, c(std\_dev, NA, NA, NA))

df3

**QUESTION 1 OUTPUT: *a)*** [1] 1.4169217 -1.5901194 -1.6540510 -1.8000713 -1.7963889 -1.6123040 -0.6487185 1.6522300 -1.0490534

[10] 1.2054332 ## unif part

[1] 6.314699 6.175024 1.341420 8.149115 4.223343 2.145544 8.008058 5.791026 4.317996 1.626505 ## norm part

[1] "F" "E" "W" "N" "N" "C" "P" "T" "T" "A" ## letter part

**b)** unif norm LET

1 1.4169217 6.314699 F

2 -1.5901194 6.175024 E

3 -1.6540510 1.341420 W

4 -1.8000713 8.149115 N

5 -1.7963889 4.223343 N

6 -1.6123040 2.145544 C

7 -0.6487185 8.008058 P

8 1.6522300 5.791026 T

9 -1.0490534 4.317996 T

10 1.2054332 1.626505 A

**c)** unif norm LET ab

1 1.4169217 6.314699 F A

2 -1.5901194 6.175024 E A

3 -1.6540510 1.341420 W A

4 -1.8000713 8.149115 N A

5 -1.7963889 4.223343 N A

6 -1.6123040 2.145544 C B

7 -0.6487185 8.008058 P B

8 1.6522300 5.791026 T B

9 -1.0490534 4.317996 T B

10 1.2054332 1.626505 A B

**d)** unif norm LET ab

1 1.4169217 6.314699 F A

8 1.6522300 5.791026 T B

10 1.2054332 1.626505 A B

4 0.2235042 NA <NA> <NA>

**##### QUESTION 2 #####**

devtools::install\_github("tidyverse/dsbox")

library(dsbox)

**### Part a: What are the three most common first names of purchasers?**

df <- data.frame(lego\_sales)

df

v1 <- c(df$first\_name)

v1

names(df)

name\_counts <- table(v1)

name\_counts

sorted\_names <- names(name\_counts)[order(-name\_counts)]

sorted\_names

top\_three\_names <- sorted\_names[1:3]

top\_three\_names

**### Part b: What are the three least common themes of Lego sets purchased?**

df2 <- data.frame(lego\_sales)

df2

v2 <- c(df2$theme)

v2

name\_counts2 <- table(v2)

name\_counts2

sorted\_names2 <- names(name\_counts2)[order(name\_counts2)]

sorted\_names2

top\_three\_names2 <- sorted\_names2[1:3]

top\_three\_names2

**### Part c: Among the most common theme of Lego sets purchased, what is the**

**#most common subtheme?**

df3 <- data.frame(lego\_sales)

most\_common\_theme <- table(lego\_sales$theme)

most\_common\_theme

most\_common\_theme <- names(most\_common\_theme[which.max(most\_common\_theme)])

most\_common\_theme # most common theme is "Star Wars”

subset(df3, select = c(theme,subtheme), subset =(theme == "Star Wars" ))

df4 <- subset(df3, select = c(theme,subtheme), subset =(theme == "Star Wars" ))

v1 <- table(df4$subtheme)

most\_common\_subtheme <- names(v1[which.max(v1)])

most\_common\_subtheme

**### Part d: Create a new variable called age\_group in this data set and**

**#group the ages into the following**

**#categories: “18 and under”, “19 - 25”, “26 - 35”, “36 - 50”, “51 and over”.**

age\_group <- cut(lego\_sales$age, breaks = c(0, 19, 26, 36, 51, 100), labels = c("18 and under", "19-25", "26-35", "36-50", "51 and over"), right = FALSE)

age\_group

lego\_sales$age\_group <- age\_group

**### Part e: Which age group has purchased the highest number of Lego sets?**

**#To this end, calculate first the sum of sets for each age group using for loop.**

**#Then identify the age group with the highest number of lego sets.**

a <- 0

b <- 0

c <- 0

d <- 0

e <- 0

for(i in 1:length(lego\_sales$number)){

if(lego\_sales$age\_group[i] == "18 and under"){

a <- a + lego\_sales$quantity[i]

} else if(lego\_sales$age\_group[i] == "19-25"){

b <- b + lego\_sales$quantity[i]

} else if(lego\_sales$age\_group[i] == "26-35"){

c <- c + lego\_sales$quantity[i]

} else if(lego\_sales$age\_group[i] == "36-50"){

d <- d + lego\_sales$quantity[i]

} else if(lego\_sales$age\_group[i] == "51 and over"){

e <- e + lego\_sales$quantity[i]}

}

age\_groups <- c("18 and under", "19-25", "26-35", "36-50", "51 and over")

total\_sales <- c(a, b, c, d, e)

sorted\_age\_groups <- age\_groups[order(total\_sales, decreasing = TRUE)]

sorted\_age\_groups[1]

**### Part f: Which age group has spent the most money on legos? Please check how**

**#aggregate function can be used to calculate the mean spending for each age**

**#group. Then identify the age group with the highest mean spending.**

total <- 0

age\_money <- function(x){

age\_filtered <- lego\_sales[lego\_sales$age\_group == x,]

unique\_people <- unique(age\_filtered[c("first\_name", "last\_name")])

for(i in 1:length(age\_filtered$number)){

total <- total + (age\_filtered$quantity[i] \* age\_filtered$us\_price[i])

}

total/nrow(unique\_people)

}

age\_money("18 and under")

age\_money("19-25")

age\_money("26-35")

age\_money("36-50")

age\_money("51 and over")

# 18 and under group has the highest mean

**QUESTION 2 OUTPUT: *a)***[1] "Jackson" "Jacob" "Joseph"

**b)** [1] "Classic" "Ideas" "Collectable Minifigures"

**c)** [1] "The Force Awakens"

**d)** [1] 19-25 26-35 26-35 36-50 36-50 36-50 19-25

[8] 19-25 36-50 36-50 19-25 19-25 19-25 36-50

[15] 36-50 36-50 36-50 36-50 36-50 36-50 36-50

[22] 36-50 36-50 36-50 36-50 36-50 36-50 26-35

[29] 26-35 26-35 26-35 51 and over 51 and over 51 and over 36-50

[36] 36-50 36-50 36-50 51 and over 51 and over 19-25 19-25

[43] 19-25 36-50 18 and under 18 and under 36-50 36-50 36-50

[50] 26-35 26-35 36-50 19-25 36-50 19-25 19-25

[57] 19-25 19-25 19-25 26-35 26-35 19-25 19-25

[64] 19-25 26-35 26-35 26-35 19-25 26-35 51 and over

[71] 18 and under 18 and under 18 and under 18 and under 18 and under 26-35 26-35

[78] 26-35 26-35 19-25 19-25 36-50 36-50 36-50

[85] 51 and over 51 and over 51 and over 19-25 19-25 19-25 36-50

[92] 36-50 19-25 19-25 19-25 36-50 36-50 36-50

[99] 36-50 36-50 19-25 19-25 19-25 19-25 26-35

[106] 36-50 36-50 36-50 36-50 36-50 51 and over 51 and over

[113] 19-25 19-25 19-25 51 and over 51 and over 51 and over 26-35

[120] 26-35 36-50 36-50 36-50 36-50 36-50 36-50

[127] 19-25 36-50 19-25 19-25 19-25 26-35 26-35

[134] 26-35 51 and over 51 and over 26-35 26-35 26-35 26-35

[141] 26-35 36-50 36-50 36-50 36-50 19-25 19-25

[148] 19-25 19-25 19-25 19-25 51 and over 51 and over 26-35

[155] 26-35 26-35 26-35 26-35 26-35 26-35 26-35

[162] 26-35 26-35 26-35 26-35 26-35 26-35 26-35

[169] 26-35 26-35 26-35 36-50 36-50 36-50 36-50

[176] 36-50 36-50 26-35 26-35 19-25 51 and over 51 and over

[183] 51 and over 19-25 26-35 26-35 26-35 18 and under 18 and under

[190] 19-25 19-25 19-25 19-25 19-25 19-25 26-35

[197] 26-35 26-35 26-35 26-35 26-35 26-35 26-35

[204] 36-50 36-50 26-35 26-35 36-50 36-50 36-50

[211] 36-50 26-35 26-35 26-35 26-35 26-35 26-35

[218] 19-25 19-25 19-25 19-25 19-25 19-25 18 and under

[225] 18 and under 18 and under 36-50 36-50 36-50 36-50 36-50

[232] 26-35 26-35 26-35 36-50 36-50 36-50 36-50

[239] 36-50 36-50 19-25 36-50 36-50 36-50 36-50

[246] 36-50 36-50 36-50 36-50 36-50 36-50 36-50

[253] 26-35 26-35 51 and over 36-50 36-50 26-35 26-35

[260] 36-50 36-50 36-50 36-50 26-35 26-35 26-35

[267] 26-35 26-35 19-25 26-35 26-35 19-25 19-25

[274] 19-25 19-25 19-25 19-25 19-25 19-25 19-25

[281] 36-50 36-50 36-50 36-50 18 and under 18 and under 18 and under

[288] 18 and under 18 and under 26-35 36-50 36-50 36-50 36-50

[295] 36-50 36-50 36-50 36-50 19-25 19-25 19-25

[302] 19-25 19-25 26-35 26-35 36-50 36-50 36-50

[309] 26-35 19-25 19-25 19-25 19-25 19-25 51 and over

[316] 51 and over 51 and over 36-50 36-50 36-50 36-50 36-50

[323] 19-25 19-25 36-50 36-50 36-50 36-50 36-50

[330] 36-50 36-50 36-50 36-50 51 and over 51 and over 51 and over

[337] 51 and over 19-25 19-25 19-25 19-25 26-35 26-35

[344] 36-50 36-50 36-50 36-50 36-50 26-35 26-35

[351] 26-35 26-35 19-25 19-25 36-50 36-50 36-50

[358] 36-50 36-50 36-50 51 and over 26-35 26-35 26-35

[365] 26-35 51 and over 51 and over 26-35 26-35 19-25 19-25

[372] 19-25 51 and over 51 and over 19-25 36-50 36-50 36-50

[379] 36-50 36-50 36-50 36-50 26-35 26-35 26-35

[386] 26-35 19-25 19-25 19-25 19-25 26-35 26-35

[393] 26-35 26-35 36-50 36-50 36-50 36-50 36-50

[400] 36-50 26-35 26-35 26-35 36-50 36-50 19-25

[407] 19-25 19-25 19-25 19-25 26-35 26-35 26-35

[414] 26-35 51 and over 51 and over 51 and over 36-50 36-50 36-50

[421] 36-50 36-50 36-50 36-50 36-50 36-50 36-50

[428] 36-50 36-50 26-35 26-35 36-50 36-50 36-50

[435] 36-50 36-50 36-50 36-50 36-50 18 and under 18 and under

[442] 18 and under 26-35 18 and under 18 and under 18 and under 26-35 26-35

[449] 26-35 26-35 26-35 26-35 26-35 26-35 26-35

[456] 51 and over 51 and over 51 and over 51 and over 51 and over 51 and over 26-35

[463] 26-35 26-35 26-35 26-35 26-35 19-25 19-25

[470] 19-25 19-25 19-25 26-35 26-35 36-50 36-50

[477] 18 and under 18 and under 26-35 26-35 26-35 26-35 36-50

[484] 36-50 36-50 36-50 36-50 51 and over 51 and over 19-25

[491] 19-25 19-25 51 and over 51 and over 51 and over 26-35 26-35

[498] 26-35 36-50 36-50 36-50 36-50 36-50 36-50

[505] 36-50 36-50 36-50 19-25 19-25 19-25 36-50

[512] 36-50 26-35 26-35 26-35 36-50 26-35 26-35

[519] 26-35 26-35 26-35 19-25 19-25 51 and over 51 and over

[526] 51 and over 19-25 19-25 26-35 26-35 36-50 36-50

[533] 36-50 36-50 36-50 36-50 36-50 51 and over 51 and over

[540] 19-25 51 and over 51 and over 26-35 26-35 26-35 26-35

[547] 51 and over 51 and over 36-50 36-50 36-50 36-50 51 and over

[554] 51 and over 36-50 36-50 36-50 26-35 26-35 26-35

[561] 26-35 26-35 26-35 36-50 36-50 36-50 36-50

[568] 36-50 26-35 26-35 19-25 19-25 19-25 36-50

[575] 36-50 26-35 26-35 26-35 26-35 26-35 26-35

[582] 19-25 19-25 19-25 19-25 26-35 26-35 26-35

[589] 26-35 26-35 19-25 19-25 19-25 19-25 51 and over

[596] 51 and over 51 and over 36-50 36-50 36-50 26-35 26-35

[603] 36-50 26-35 26-35 26-35 26-35 26-35 19-25

[610] 19-25 26-35 18 and under 18 and under 18 and under 18 and under 18 and under

[617] 26-35 26-35 36-50 36-50

Levels: 18 and under 19-25 26-35 36-50 51 and over

**e)** [1] "36-50"

**f)** > age\_money("18 and under")

[1] 86.23182

> age\_money("19-25")

[1] 91.46778

> age\_money("26-35")

[1] 101.0177

> age\_money("36-50")

[1] 114.8539

> age\_money("51 and over")

[1] 91.66963

#36-50 has the highest mean

**##### QUESTION 3 #####**

**### Obtain the following output:**

## [1] "Fizz"

## [1] "Buzz"

## [1] "Fizz"

## [1] "Buzz"

## [1] "Fizz"

**### Part a: Using while loop**

a <- 1

while(a <= 5){

if(a %% 2 ==1){

print("Fizz")

a <- a+1

}else{

print("Buzz")

a <- a+1

}

}

}**### Part b: Using repeat command**

b <- 1

repeat {

if (b > 5) {

break

}

if (b %% 2 == 1) {

print("Fizz")

} else if (b %% 2 == 0) {

print("Buzz")

}

b <- b + 1

}

**### Part c: Using for loop**

m <- c(1,2,3,4,5)

for(i in m){

if(i %% 2 == 1){

print("Fizz")

}else{

print("Buzz")

}

}

}}**QUESTION 3 OUTPUT: a)** [1] "Fizz"

[1] "Buzz"

[1] "Fizz"

[1] "Buzz"

[1] "Fizz"

**b** [1] "Fizz"

[1] "Buzz"

[1] "Fizz"

[1] "Buzz"

[1] "Fizz"

**c)** [1] "Fizz"

[1] "Buzz"

[1] "Fizz"

[1] "Buzz"

[1] "Fizz"