



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO®

Instituto Tecnológico de Tijuana

Carrera:

Ingeniería Informática

Nombre de la Materia: MINERÍA DE DATOS

Actividad: Práctica 1

Profesor: JOSE CHRISTIAN ROMERO HERNANDEZ

Alumno(s): Díaz Ruiz Uriel 18210839

Pérez Ortega Victoria Valeria 18210718

Fecha de entrega: 07/03/22

Pruebe la ley de los números grandes para N números aleatorios normalmente distribuidos con media = 0, stdev = 1:

Cree un script R que cuente cuántos de estos números caen entre -1 y 1 y divida por la cantidad total de N

Sabes que $E(X) = 68.2\%$

Compruebe que $\text{Mean}(X_n) \rightarrow E(X)$ al volver a ejecutar el script mientras aumenta N

Indirecta:

1. Inicializar el tamaño de la muestra
2. Inicializar contador
3. bucle para(i en rnorm(tamaño))
4. Compruebe si la variable iterada cae
5. Aumentar el contador si la condición es verdadera
6. devolver un resultado <- contador / N

```
# **Ley de los números grandes para N números Aleatorios  
Normalmente distribuidos con la media = 0, stdev = 1.**
```

```
  Cree un script de R que cuente de cuántos de estos números se encuentran entre  
-1 y 1 y lo dividirá por la cantidad total de N
```

```
Sabes que: ** E (X) = 68,2% **
```

```
  Compruebe que Sean  $(X_n) \rightarrow E(X)$  mientras vuelve a ejecutar su script  
mientras aumenta N
```

```
# ** Sugerencias: **
```

1. Inicializar el tamaño de la muestra
2. Inicializar contador
3. bucle para(i en rnorm(tamaño))

```

4. Compruebe si la variable iterada cae
5. Aumentar el contador si la condición es verdadera
6. devolver un resultado <- contador / N
# **Código paso a paso**
## 1. Inicializar el tamaño de la muestra con 10
x <- rnorm(10)
y <- dnorm(x, mean = 0, sd = 1)
plot(x,y)
## 2. Inicializar contador y la variable del resultado
count <- 0
res <- 0
## 3. Ciclo for que recorre la muestra y aumenta el contador a la vez que
imprime la cuenta
for(i in rnorm(10))
{
  count <- count + 1
  print(count)
  ## 4. Verifique si la variable iterada cae entre -1 and 1, si es así
  entonces lo imprime
  if(i <= 1 && i >= -1){
    print(i)
    print("It's between -1 y 1")

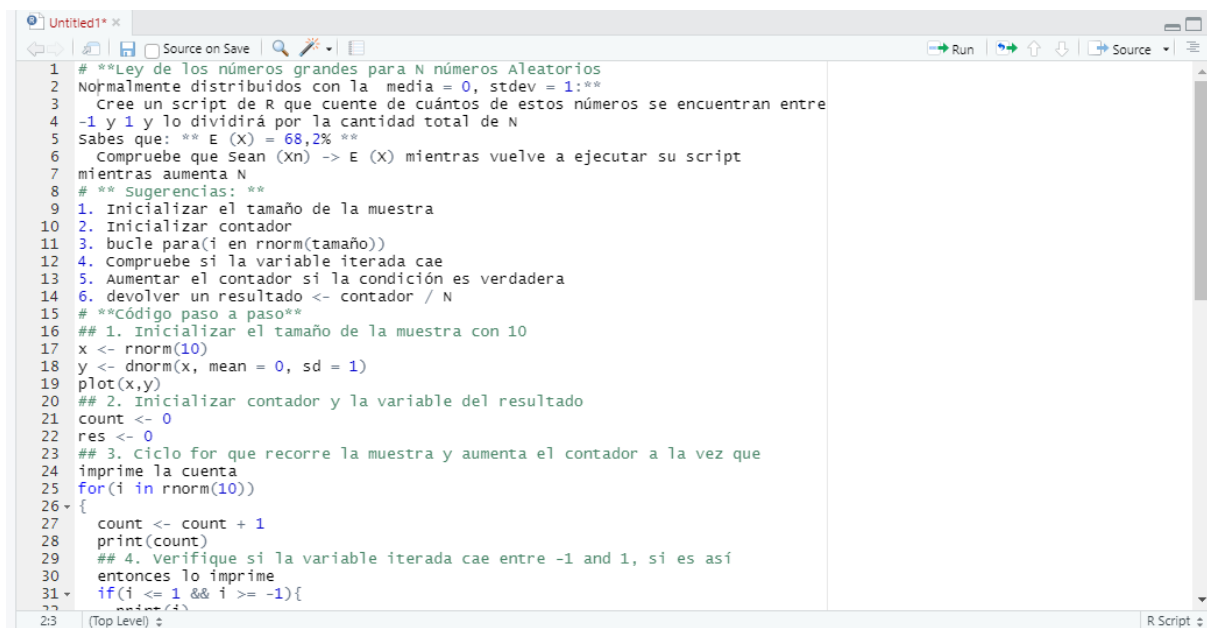
    ## 5. Aumente el contador si la condición es verdadera, de lo contrario
    imprime que el número está fuera del rango
    res <- res + 1
  }
  else
  {
    print(i)
    print("Out of range")
  }
}
## 6. Devuelve un resultado y lo divide entre el contador, al final imprime
el resultado
Res <- res / count
print(Res)
# **Código completo**
x <- rnorm(10)
y <- dnorm(x, mean = 0, sd = 1)
plot(x,y)
count <- 0
res <- 0
for(i in rnorm(10))
{
  count <- count + 1
  print(count)
  if(i <= 1 && i >= -1){

```

```

print(i)
print("It's between -1 y 1")
res <- res + 1
}
else
{
  print(i)
  print("Out of range")
}
}
Res <- res / count
print(Res)

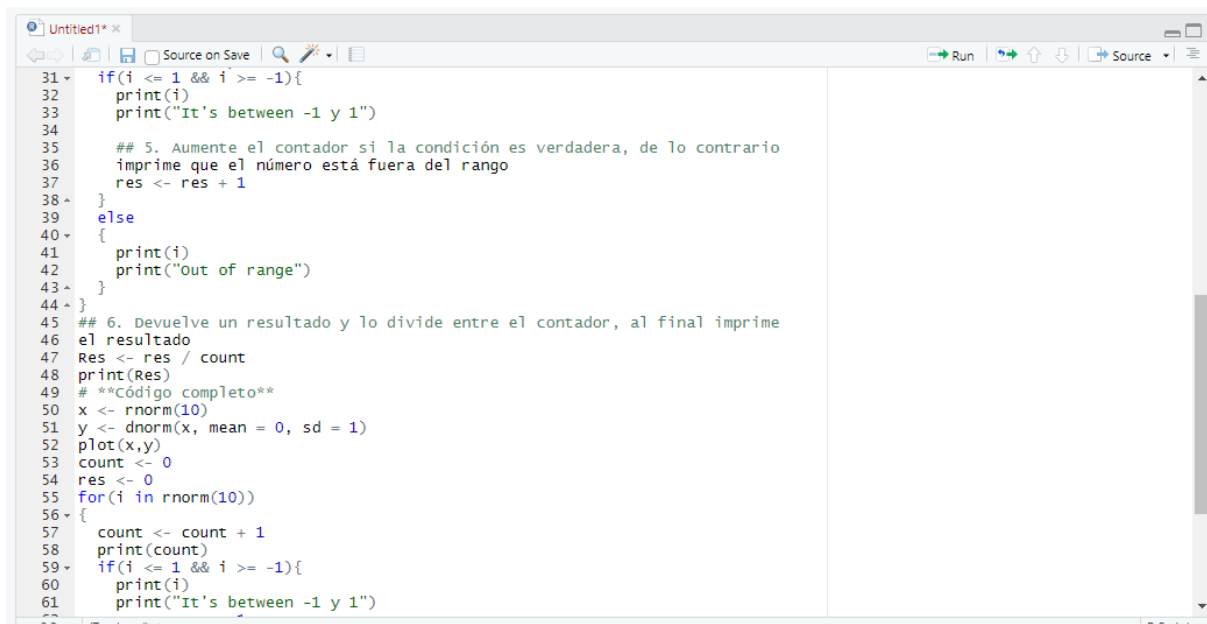
```



```

1  # **Ley de los números grandes para N números Aleatorios
2  Normalmente distribuidos con la media = 0, stdev = 1:**
3  Cree un script de R que cuente de cuántos de estos números se encuentran entre
4  -1 y 1 y lo dividirá por la cantidad total de N
5  Sabes que: ** E (X) = 68,2% **
6  Compruebe que Sean (Xn) -> E (X) mientras vuelve a ejecutar su script
7  mientras aumenta N
8  # ** sugerencias: **
9  1. Inicializar el tamaño de la muestra
10 2. Inicializar contador
113. bucle para(i en rnorm(tamaño))
124. Compruebe si la variable iterada cae
135. Aumentar el contador si la condición es verdadera
146. devolver un resultado <- contador / N
15# **Código paso a paso**
16## 1. Inicializar el tamaño de la muestra con 10
17x <- rnorm(10)
18y <- dnorm(x, mean = 0, sd = 1)
19plot(x,y)
20## 2. Inicializar contador y la variable del resultado
21count <- 0
22res <- 0
23## 3. Ciclo for que recorre la muestra y aumenta el contador a la vez que
24imprime la cuenta
25for(i in rnorm(10))
26{
27  count <- count + 1
28  print(count)
29  ## 4. Verifique si la variable iterada cae entre -1 and 1, si es así
30  entonces lo imprime
31  if(i <= 1 && i >= -1){
32    print(i)
33    print("It's between -1 y 1")
34  }
35  ## 5. Aumente el contador si la condición es verdadera, de lo contrario
36  imprime que el número está fuera del rango
37  res <- res + 1
38  }
39  else
40  {
41    print(i)
42    print("Out of range")
43  }
44  }
45  ## 6. Devuelve un resultado y lo divide entre el contador, al final imprime
46  el resultado
47  Res <- res / count
48  print(Res)
49  # **Código completo**
50  x <- rnorm(10)
51  y <- dnorm(x, mean = 0, sd = 1)
52  plot(x,y)
53  count <- 0
54  res <- 0
55  for(i in rnorm(10))
56  {
57    count <- count + 1
58    print(count)
59    if(i <= 1 && i >= -1){
60      print(i)
61      print("It's between -1 y 1")
62    }
63    else
64    {
65      print(i)
66      print("Out of range")
67    }
68  }
69  Res <- res / count
70  print(Res)

```



```

31  if(i <= 1 && i >= -1){
32    print(i)
33    print("It's between -1 y 1")
34  }
35  ## 5. Aumente el contador si la condición es verdadera, de lo contrario
36  imprime que el número está fuera del rango
37  res <- res + 1
38  }
39  else
40  {
41    print(i)
42    print("Out of range")
43  }
44  }
45  ## 6. Devuelve un resultado y lo divide entre el contador, al final imprime
46  el resultado
47  Res <- res / count
48  print(Res)
49  # **Código completo**
50  x <- rnorm(10)
51  y <- dnorm(x, mean = 0, sd = 1)
52  plot(x,y)
53  count <- 0
54  res <- 0
55  for(i in rnorm(10))
56  {
57    count <- count + 1
58    print(count)
59    if(i <= 1 && i >= -1){
60      print(i)
61      print("It's between -1 y 1")
62    }
63    else
64    {
65      print(i)
66      print("Out of range")
67    }
68  }
69  Res <- res / count
70  print(Res)

```

```
43 ^ }
44 ^ }
45 ## 6. Devuelve un resultado y lo divide entre el contador, al final imprime
46 el resultado
47 Res <- res / count
48 print(Res)
49 # **Código completo**
50 x <- rnorm(10)
51 y <- dnorm(x, mean = 0, sd = 1)
52 plot(x,y)
53 count <- 0
54 res <- 0
55 for(i in rnorm(10))
56 {
57   count <- count + 1
58   print(count)
59   if(i <= 1 && i >= -1){
60     print(i)
61     print("It's between -1 y 1")
62     res <- res + 1
63   }
64   else
65   {
66     print(i)
67     print("out of range")
68   }
69 }
70 Res <- res / count
71 print(Res)
72
73
2:3 | (Top Level) ↕ R Script
```

Environment	History	Connections	Tutorial
Import Dataset 136 MiB			
R Global Environment			
values			
count	10		
i	-1.59281695071588		
res	8		
Res	0.8		
x	num [1:10] 1.611 1.213 -0.291 -1.647 1.131 ...		
y	num [1:10] 0.109 0.191 0.382 0.103 0.21 ...		