

Tarea 2-Ejercicio de programación

Escribe tu nombre aquí

05/10/2020-16/10/2020

1. Simula 50 tiros de una moneda justa.

```
set.seed(1)
tiros <- sample (0:1,50, replace = TRUE)
tiros

## [1] 0 1 0 0 1 0 0 0 1 1 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 1 1 1 0 1 0 0 1 0
## [39] 1 1 1 1 0 1 1 1 1 1 0 0
```

2. Imprime un data frame con el número de veces que salieron sol y águila en el experimento, además debe indicar la proporción de veces que salió sol.

```
Resultados <- data.frame("Número de soles"= sum(tiros),
                        "Número de águilas"= 50-sum(tiros),
                        "Proporción de soles" = sum(tiros)/50 )
Resultados

##      Número.de.soles Número.de.águilas Proporción.de.soles
## 1                23                27                0.46
```

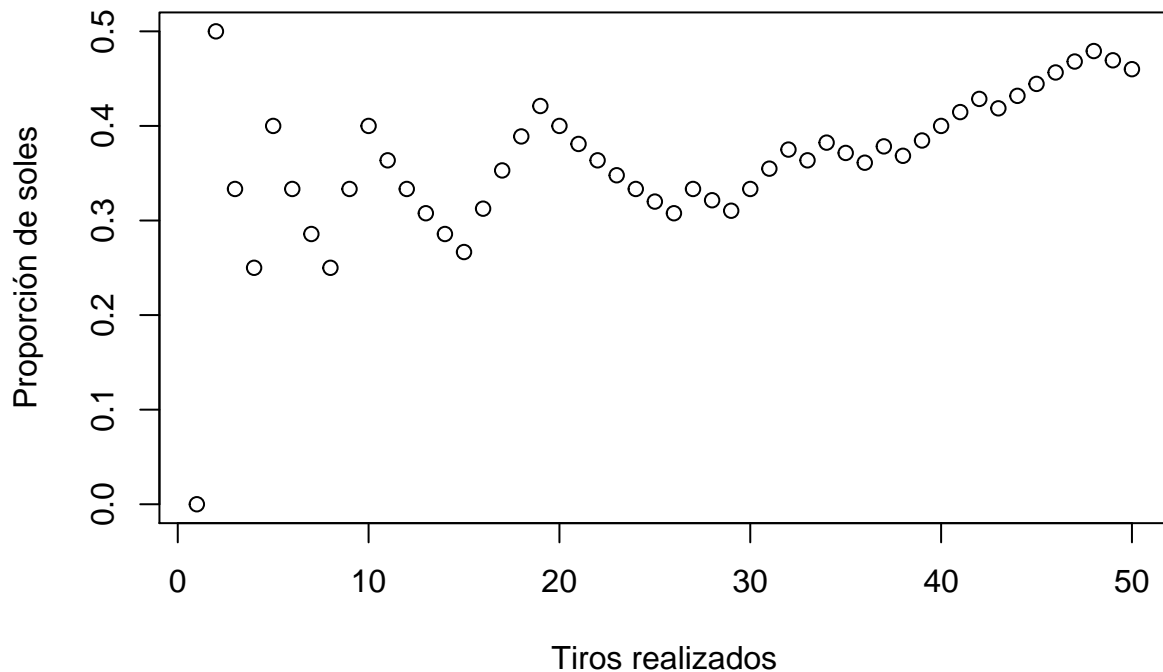
3. Grafica cómo va cambiando la proporción de soles tiro a tiro.

```
tiro_por_tiro <- data.frame("Tiro" = vector("integer"),
                          "Número de soles"= vector(),
                          "Proporción de soles" = vector() )
for (i in 1:50) {
  tiro_actual <- tiros[1:i]
  datos_nuevos <- data.frame("Tiro" = i,
                            "Número de soles"= sum(tiro_actual),
                            "Proporción de soles" =
                              sum(tiro_actual)/ i)
  tiro_por_tiro <- rbind(tiro_por_tiro, datos_nuevos)
}
tiro_por_tiro

##      Tiro Número.de.soles Proporción.de.soles
## 1      1                0      0.0000000
## 2      2                1      0.5000000
## 3      3                1      0.3333333
## 4      4                1      0.2500000
## 5      5                2      0.4000000
## 6      6                2      0.3333333
```

## 7	7	2	0.2857143
## 8	8	2	0.2500000
## 9	9	3	0.3333333
## 10	10	4	0.4000000
## 11	11	4	0.3636364
## 12	12	4	0.3333333
## 13	13	4	0.3076923
## 14	14	4	0.2857143
## 15	15	4	0.2666667
## 16	16	5	0.3125000
## 17	17	6	0.3529412
## 18	18	7	0.3888889
## 19	19	8	0.4210526
## 20	20	8	0.4000000
## 21	21	8	0.3809524
## 22	22	8	0.3636364
## 23	23	8	0.3478261
## 24	24	8	0.3333333
## 25	25	8	0.3200000
## 26	26	8	0.3076923
## 27	27	9	0.3333333
## 28	28	9	0.3214286
## 29	29	9	0.3103448
## 30	30	10	0.3333333
## 31	31	11	0.3548387
## 32	32	12	0.3750000
## 33	33	12	0.3636364
## 34	34	13	0.3823529
## 35	35	13	0.3714286
## 36	36	13	0.3611111
## 37	37	14	0.3783784
## 38	38	14	0.3684211
## 39	39	15	0.3846154
## 40	40	16	0.4000000
## 41	41	17	0.4146341
## 42	42	18	0.4285714
## 43	43	18	0.4186047
## 44	44	19	0.4318182
## 45	45	20	0.4444444
## 46	46	21	0.4565217
## 47	47	22	0.4680851
## 48	48	23	0.4791667
## 49	49	23	0.4693878
## 50	50	23	0.4600000

```
plot(tiro_por_tiro[,1],
     tiro_por_tiro[,3],
     xlab = "Tiros realizados",
     ylab = "Proporción de soles")
```

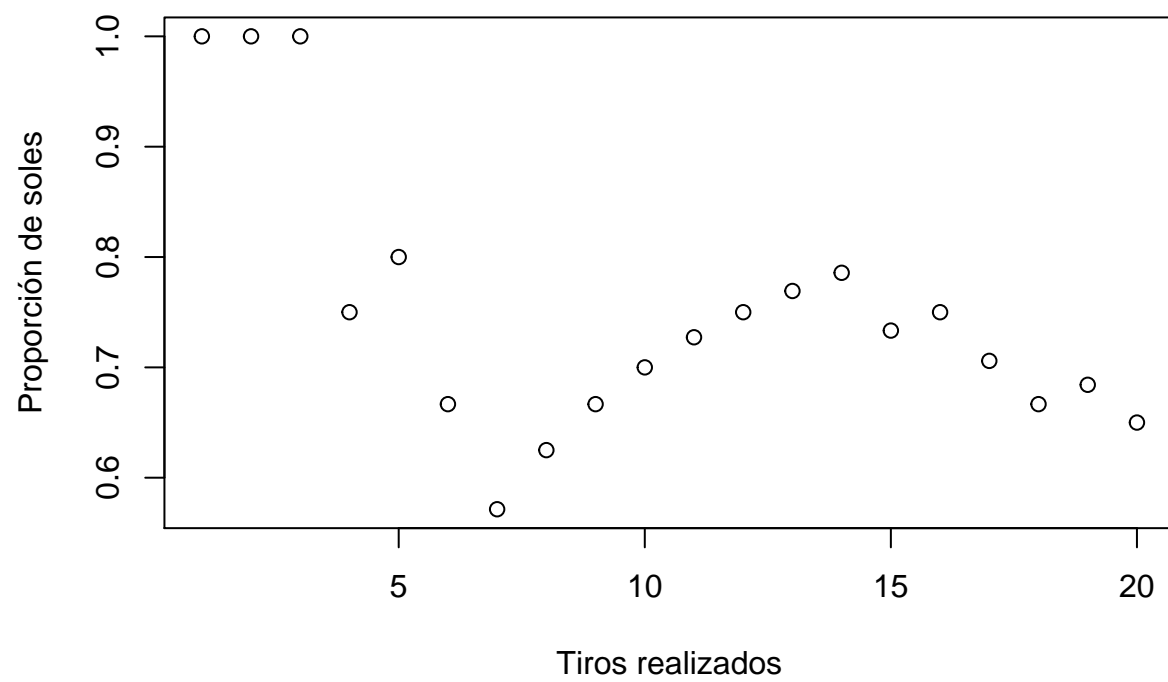


4. Toma una moneda no justa. Grafica la proporción de soles que salen primero en 20 tiros y luego en 1000 tiros. Tip: si haces una función esto puede resultar más fácil.

```
Prop_soles <- function(num_tiros, proba = .5){
  set.seed(1)
  tiros <- sample(0:1, num_tiros, replace = TRUE,
                 prob = c(1-proba, proba))
  tiro_por_tiro <- data.frame("Tiro" = vector("integer"),
                             "Número de soles" = vector(),
                             "Proporción de soles" = vector() )

  for (i in 1:num_tiros) {
    tiro_actual <- tiros[1:i]
    datos_nuevos <- data.frame("Tiro" = i, "Número de soles" =
                              sum(tiro_actual), "Proporción de
                              soles" = sum(tiro_actual)/ i)
    tiro_por_tiro <- rbind(tiro_por_tiro, datos_nuevos)
  }
  tiro_por_tiro
  plot(tiro_por_tiro[,1], tiro_por_tiro[,3],
       xlab = "Tiros realizados",
       ylab = "Proporción de soles")
}

Prop_soles(20, .7)
```



```
Prop_soles(1000, .7)
```

