

# Practica4

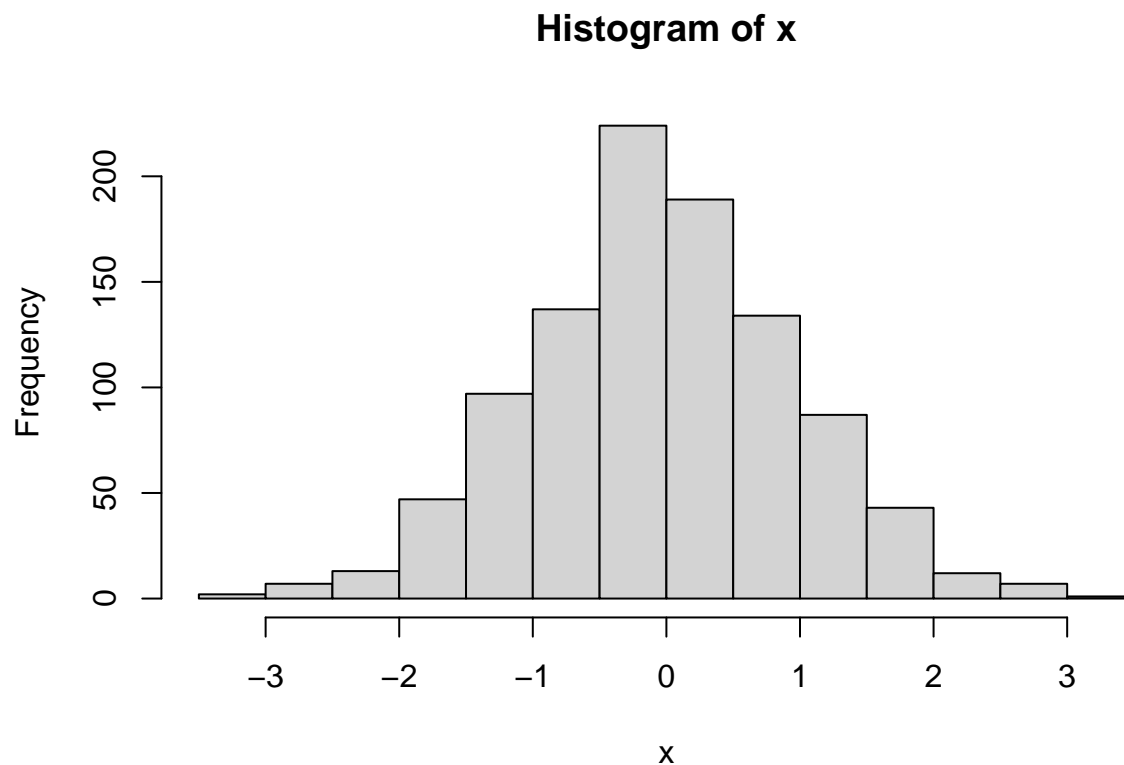
Carlos Mota Romero

2023-03-13

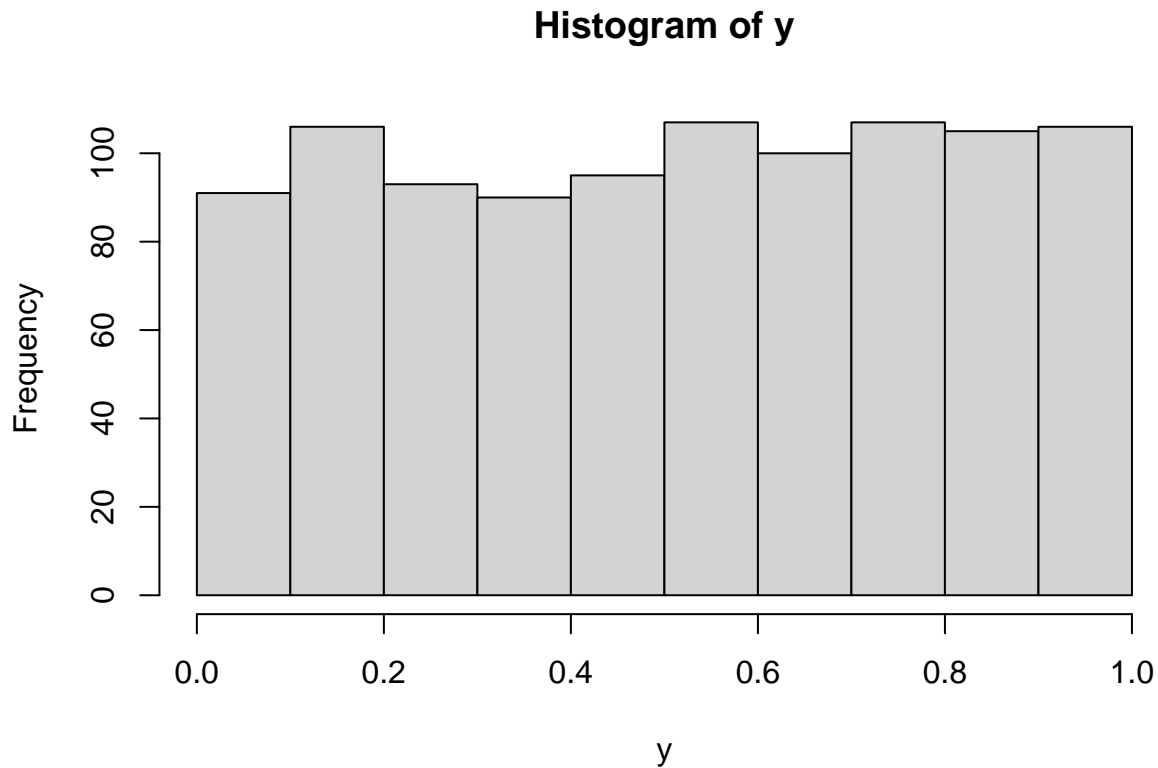
## Ejercicio 1

Queremos generar una serie de números aleatorios; si lo generamos con funciones como `rnorm` o `runif` no se crean valores verdaderamente aleatorios, sino que se generan mediante patrones predefinidos. Es esta la razón por la que los gráficos que generamos tendrán siempre la misma forma si utilizamos la misma función.

```
x <- rnorm(1000)
hist(x)
```



```
y <- runif(1000)
hist(y)
```



Si queremos generar valores aleatorios, pero reproducibles, lo que haremos será utilizar la función `set.seed()` que le damos un valor (1, 2, 3, etc.) y utilizando ahora las funciones anteriores nos generará patrones de números aleatorios que son reproducibles.

```
set.seed(1)
rnorm(5)
```

```
## [1] -0.6264538  0.1836433 -0.8356286  1.5952808  0.3295078
```

```
rnorm(10)
```

```
## [1] -0.8204684  0.4874291  0.7383247  0.5757814 -0.3053884  1.5117812
## [7]  0.3898432 -0.6212406 -2.2146999  1.1249309
```

```
set.seed(1)
rnorm(5)
```

```
## [1] -0.6264538  0.1836433 -0.8356286  1.5952808  0.3295078
```

#Ejercicio 2: Shapiro test. Shapiro test comprueba si la población de datos tiene una distribución normal. Esto se mira con el p value, si es superior a 0'05 indica que la población es probablemente normal. Si no lo fuera, significaría que tiene una distribución distinta a la normal.

Para comprobar esto generamos dos tipos de distribuciones, una normal con la función `rnorm` y una exponencial con la función `rexp`. Al correr la función `shapiro.test`, nos da unos valores p para cada conjunto

de valores. Podemos comprobar que la población de datos generada con `rnorm` tiene un valor p mayor que 0'05, mientras que la generada con `rexp` tiene un valor de p muy bajo y por tanto no tendrá una distribución normal.

```
x <- rnorm(1000)
shapiro.test(x)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.99876, p-value = 0.7262
```

```
z <- rexp(1000)
shapiro.test(z)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  z
## W = 0.82106, p-value < 2.2e-16
```