

Ejercicios

Uriel Paluch

10/11/2021

Metodos

```
newtonCotesCerradas <- function(limiteInferior, limiteSuperior, funcion, n){  
  h <- (limiteSuperior - limiteInferior)/n  
  
  fx <- rep(NA, times = (n+1))  
  for (i in 1:(n+1)) {  
    fx[i] <- eval(funcion, list(x = limiteInferior + (i-1)*h))  
  }  
  
  if (n == 1){  
    return((h/2) * (fx[1] + fx[2]))  
  }  
  else if (n == 2){  
    return((h/3) * (fx[1] + 4*fx[2] + fx[3]))  
  }  
  else if(n == 3){  
    return((3/8)*h*(fx[1] + 3*fx[2] + 3*fx[3] + fx[4]))  
  }  
  else if(n == 4){  
    return((2/45) * h * ( 7 * fx[1] + 32 * fx[2] + 12 * fx[3] + 32 * fx[4] + 7 * fx[5]))  
  }  
}
```

```
newtonCotesAbiertas <- function(limiteInferior, limiteSuperior, funcion, n){  
  h <- (limiteSuperior - limiteInferior)/(n+2)  
  
  fx <- rep(NA, times = (n+1))  
  for (i in 1:(n+1)) {  
    fx[i] <- eval(funcion, list(x = limiteInferior + i*h))  
  }  
  
  if (n == 0){  
    return(2 * h * fx[1])  
  }  
  else if (n == 1){  
    return((3/2)* h * (fx[1] + fx[2]))  
  }  
  else if(n == 2){  
    return((4/3)*h*(2*fx[1] - fx[2] + 2*fx[3]))  
  }  
}
```

```

else if(n == 3){
  return((5/24) * h * ( 11 * fx[1] + fx[2] + fx[3] + 11 * fx[4]))
}
}

PuntoMedioCompuesta <- function(limiteInferior, limiteSuperior, funcion, n){
  h <- (limiteSuperior - limiteInferior)/(n + 2)

  suma <- 0

  x <- rep(NA, times = (n+2))
  for (i in -1:(n+1)) {
    x[i+2] <- limiteInferior + (i + 1) * h
  }

  for (j in 1:(n/2+1)) {
    suma <- suma + eval(funcion, list(x = x[2*j]))
  }
  return(2 * h * suma)
}

IntegracionCompuesta <- function(limiteInferior, limiteSuperior, funcion, n, cantIntervalos){
  #browser()
  if ((n == 2 || n == 0) && cantIntervalos%%2 != 0){
    return("cantIntervalos debe ser un entero par")
  }

  cantIntervalos <- cantIntervalos/n

  crecimientoIntervalo <- (limiteSuperior-limiteInferior)/cantIntervalos

  fx <- rep(NA, times = (n+1))

  resultado <- 0

  for (i in 1:cantIntervalos) {
    limiteSuperior <- limiteInferior + crecimientoIntervalo

    if (n != 0){
      h <- (limiteSuperior - limiteInferior)/n
    }

    for (i in 1:(n+1)) {
      fx[i] <- eval(funcion, list(x = limiteInferior + (i-1)*h))
    }

    # Trapecio
    if (n == 1){
      resultado <- resultado + (h/2) * (fx[1] + fx[2])
    }
  }
}

```

```

#Simpson
else if(n == 2){
  resultado <- resultado + (h/3) * (fx[1] + 4*fx[2] + fx[3])
}

limiteInferior <- limiteSuperior
}

return(resultado)
}

```

Ejercicio 1

Regla del trapecio

```

# n = 1. Regla del trapecio.
# n = 2. Regla de Simpson.
# n = 3. Regla de tres octavos de Simpson.
# n = 4 regla de NC cerrada con n = 4.
# Poner la funcion con "x" como incognita
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 0.5, funcion = expression(2/(x-4)), n = 1)

```

a

```
## [1] -0.2678571
```

```
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 0.5, funcion = expression(2/(x-4)), n = 2)
```

```
## [1] -0.2670635
```

```
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 0.5, funcion = expression(2/(x-4)), n = 3)
```

```
## [1] -0.2670631
```

```
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 0.5, funcion = expression(2/(x-4)), n = 4)
```

```
## [1] -0.2670628
```

```

# n = 1. Regla del trapecio.
# n = 2. Regla de Simpson.
# n = 3. Regla de tres octavos de Simpson.
# n = 4 regla de NC cerrada con n = 4.
# Poner la funcion con "x" como incognita
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 0.35, funcion = expression(2/(x^2-4)), n = 1)

```

b

```
## [1] -0.1777643
```

```
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 0.35, funcion = expression(2/(x^2-4)), n = 2)
```

```
## [1] -0.1768216
```

```
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 0.35, funcion = expression(2/(x^2-4)), n = 3)
```

```
## [1] -0.1768207
```

```
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 0.35, funcion = expression(2/(x^2-4)), n = 4)

## [1] -0.17682
```

Regla de Simpson

```
# n = 1. Regla del trapecio.
# n = 2. Regla de Simpson.
# n = 3. Regla de tres octavos de Simpson.
# n = 4 regla de NC cerrada con n = 4.
# Poner la funcion con "x" como incognita
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 1, funcion = expression((x^2)*exp(-x)), n = 1)
```

c

```
## [1] 0.1839397
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 1, funcion = expression((x^2)*exp(-x)), n = 2)

## [1] 0.1624017
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 1, funcion = expression((x^2)*exp(-x)), n = 3)

## [1] 0.1614099
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = 1, funcion = expression((x^2)*exp(-x)), n = 4)

## [1] 0.1606105
```

d Este me da como el orto

```
# n = 1. Regla del trapecio.
# n = 2. Regla de Simpson.
# n = 3. Regla de tres octavos de Simpson.
# n = 4 regla de NC cerrada con n = 4.
# Poner la funcion con "x" como incognita
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = pi*4, funcion = expression(exp(3*x)*sin(2*x)),

## [1] -145.1385
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = pi*4, funcion = expression(exp(3*x)*sin(2*x)),

## [1] -48.3795
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = pi*4, funcion = expression(exp(3*x)*sin(2*x)),

## [1] -335568417388
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = pi*4, funcion = expression(exp(3*x)*sin(2*x)),

## [1] -22.58335
```

Regla de Newton-Cotes cerrada con n = 4

```
# n = 1. Regla del trapecio.
# n = 2. Regla de Simpson.
# n = 3. Regla de tres octavos de Simpson.
# n = 4 regla de NC cerrada con n = 4.
```

```
# Poner la funcion con "x" como incognita
newtonCotesCerradas(limiteInferior = 0.5, limiteSuperior = 1, funcion = expression(x^4), n = 1)
```

e

```
## [1] 0.265625
```

```
newtonCotesCerradas(limiteInferior = 0.5, limiteSuperior = 1, funcion = expression(x^4), n = 2)
```

```
## [1] 0.1940104
```

```
newtonCotesCerradas(limiteInferior = 0.5, limiteSuperior = 1, funcion = expression(x^4), n = 3)
```

```
## [1] 0.1938657
```

```
newtonCotesCerradas(limiteInferior = 0.5, limiteSuperior = 1, funcion = expression(x^4), n = 4)
```

```
## [1] 0.19375
```

f Este me da mal

```
# n = 1. Regla del trapecio.
```

```
# n = 2. Regla de Simpson.
```

```
# n = 3. Regla de tres octavos de Simpson.
```

```
# n = 4 regla de NC cerrada con n = 4.
```

```
# Poner la funcion con "x" como incognita
```

```
newtonCotesCerradas(limiteInferior = 1, limiteSuperior = 1.6, funcion = expression(2*x/(x^2-4)), n = 1)
```

```
## [1] -0.8666667
```

```
newtonCotesCerradas(limiteInferior = 1, limiteSuperior = 1.6, funcion = expression(2*x/(x^2-4)), n = 2)
```

```
## [1] -0.7391053
```

```
newtonCotesCerradas(limiteInferior = 1, limiteSuperior = 1.6, funcion = expression(2*x/(x^2-4)), n = 3)
```

```
## [1] -0.7364277
```

```
newtonCotesCerradas(limiteInferior = 1, limiteSuperior = 1.6, funcion = expression(2*x/(x^2-4)), n = 4)
```

```
## [1] -0.7341567
```

Regla de Newton-Cotes cerrada con $n = 4$

```
# n = 1. Regla del trapecio.
```

```
# n = 2. Regla de Simpson.
```

```
# n = 3. Regla de tres octavos de Simpson.
```

```
# n = 4 regla de NC cerrada con n = 4.
```

```
# Poner la funcion con "x" como incognita
```

```
newtonCotesCerradas(limiteInferior = 1, limiteSuperior = 1.5, funcion = expression(x^2*log(x)), n = 4)
```

g

```
## [1] 0.1922593
```

```
# n = 1. Regla del trapecio.
```

```
# n = 2. Regla de Simpson.
```

```
# n = 3. Regla de tres octavos de Simpson.
```

```
# n = 4 regla de NC cerrada con n = 4.
# Poner la funcion con "x" como incognita
newtonCotesCerradas(limiteInferior = 0, limiteSuperior = pi/4, funcion = expression(x*sin(x)), n = 4)

h

## [1] 0.1517469
```

Ejercicio 2

```
# n = 0. Regla del punto medio.
# n = 1.
# n = 2.
# n = 3.
# Poner la funcion con "x" como incognita
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = 0.1, funcion = expression(sqrt(x+1)), n = 0)
```

```
a

## [1] 0.1024695
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = 0.1, funcion = expression(sqrt(x+1)), n = 1)

## [1] 0.1024663
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = 0.1, funcion = expression(sqrt(x+1)), n = 2)

## [1] 0.1024598
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = 0.1, funcion = expression(sqrt(x+1)), n = 3)

## [1] 0.1024598
```

```
# n = 0. Regla del punto medio.
# n = 1.
# n = 2.
# n = 3.
# Poner la funcion con "x" como incognita
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = pi/2, funcion = expression(sin(x)^2), n = 0)
```

```
b

## [1] 0.7853982
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = pi/2, funcion = expression(sin(x)^2), n = 1)

## [1] 0.7853982
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = pi/2, funcion = expression(sin(x)^2), n = 2)

## [1] 0.7853982
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = pi/2, funcion = expression(sin(x)^2), n = 3)

## [1] 0.7853982
```

```

# n = 0. Regla del punto medio.
# n = 1.
# n = 2.
# n = 3.
# Poner la funcion con "x" como incognita
newtonCotesAbiertas(limiteInferior = 1.1, limiteSuperior = 1.5, funcion = expression(exp(x)), n = 0)

c

## [1] 1.467719
newtonCotesAbiertas(limiteInferior = 1.1, limiteSuperior = 1.5, funcion = expression(exp(x)), n = 1)

## [1] 1.470981
newtonCotesAbiertas(limiteInferior = 1.1, limiteSuperior = 1.5, funcion = expression(exp(x)), n = 2)

## [1] 1.477512
newtonCotesAbiertas(limiteInferior = 1.1, limiteSuperior = 1.5, funcion = expression(exp(x)), n = 3)

## [1] 1.477515

```

```

# n = 0. Regla del punto medio.
# n = 1.
# n = 2.
# n = 3.
# Poner la funcion con "x" como incognita
newtonCotesAbiertas(limiteInferior = 1, limiteSuperior = 10, funcion = expression(1/x), n = 0)

```

```

d

## [1] 1.636364
newtonCotesAbiertas(limiteInferior = 1, limiteSuperior = 10, funcion = expression(1/x), n = 1)

## [1] 1.767857
newtonCotesAbiertas(limiteInferior = 1, limiteSuperior = 10, funcion = expression(1/x), n = 2)

## [1] 2.074893
newtonCotesAbiertas(limiteInferior = 1, limiteSuperior = 10, funcion = expression(1/x), n = 3)

## [1] 2.116379

```

```

# n = 0. Regla del punto medio.
# n = 1.
# n = 2.
# n = 3.
# Poner la funcion con "x" como incognita
newtonCotesAbiertas(limiteInferior = 1, limiteSuperior = 5.5, funcion = expression(1/x), n = 0) + newton

```

```

e

## [1] 1.965261

```

```

newtonCotesAbiertas(limiteInferior = 1, limiteSuperior = 5.5, funcion = expression(1/x), n = 1) + newton
## [1] 2.048634
newtonCotesAbiertas(limiteInferior = 1, limiteSuperior = 5.5, funcion = expression(1/x), n = 2) + newton
## [1] 2.233251
newtonCotesAbiertas(limiteInferior = 1, limiteSuperior = 5.5, funcion = expression(1/x), n = 3) + newton
## [1] 2.249001

```

```

# n = 0. Regla del punto medio.
# n = 1.
# n = 2.
# n = 3.
# Poner la funcion con "x" como incognita
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = 1, funcion = expression(x^(1/3)), n = 0)

f
## [1] 0.7937005
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = 1, funcion = expression(x^(1/3)), n = 1)

## [1] 0.7834709
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = 1, funcion = expression(x^(1/3)), n = 2)

## [1] 0.7611137
newtonCotesAbiertas(limiteInferior = 0, limiteSuperior = 1, funcion = expression(x^(1/3)), n = 3)

## [1] 0.7593572

```

Ejercicio 3

```

# n = 1. Trapecio
# n = 2. Simpson
IntegracionCompuesta(limiteInferior = 1, limiteSuperior = 2, funcion = expression(x*log(x)), cantInterv
a
## [1] 0.6399005
IntegracionCompuesta(limiteInferior = 1, limiteSuperior = 2, funcion = expression(x*log(x)), cantInterv
## [1] 0.6363098
PuntoMedioCompuesta(limiteInferior = 1, limiteSuperior = 2, funcion = expression(x*log(x)), n = 4)

## [1] 0.6330964

```

```

# n = 1. Trapecio
# n = 2. Simpson
IntegracionCompuesta(limiteInferior = -2, limiteSuperior = 2, funcion = expression(x^3*exp(x)), cantInt
b

```



```
## [1] 31.36529
```

```
IntegracionCompuesta(limiteInferior = -2, limiteSuperior = 2, funcion = expression(x^3*exp(x)), cantInt
```

```
## [1] 22.47713
```

```
PuntoMedioCompuesta(limiteInferior = -2, limiteSuperior = 2, funcion = expression(x^3*exp(x)), n = 4)
```

```
## [1] 11.15677
```

```
# n = 1. Trapecio
```

```
# n = 2. Simpson
```

```
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 2, funcion = expression(2/(x^2+4)), cantInter
```

c

```
## [1] 0.7842408
```

```
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 2, funcion = expression(2/(x^2+4)), cantInter
```

```
## [1] 0.7853979
```

```
PuntoMedioCompuesta(limiteInferior = 0, limiteSuperior = 2, funcion = expression(2/((x^2)+4)), n = 6)
```

```
## [1] 0.7867001
```

```
# n = 1. Trapecio
```

```
# n = 2. Simpson
```

```
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = pi, funcion = expression(x^2*cos(x)), cantInt
```

d

```
## [1] -6.428722
```

```
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = pi, funcion = expression(x^2*cos(x)), cantInt
```

```
## [1] -6.274868
```

```
PuntoMedioCompuesta(limiteInferior = 0, limiteSuperior = pi, funcion = expression(x^2*cos(x)), n = 6)
```

```
## [1] -6.112736
```

```
# n = 1. Trapecio
```

```
# n = 2. Simpson
```

```
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 2, funcion = expression(exp(2*x)*sin(3*x)), c
```

e

```
## [1] -13.57598
```

```
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 2, funcion = expression(exp(2*x)*sin(3*x)), c
```

```
## [1] -14.18334
```

```
PuntoMedioCompuesta(limiteInferior = 0, limiteSuperior = 2, funcion = expression(exp(2*x)*sin(3*x)), n :
```

```
## [1] -14.99848
```

```

# n = 1. Trapecio
# n = 2. Simpson
IntegracionCompuesta(limiteInferior = 1, limiteSuperior = 3, funcion = expression(x/(x^2+4)), cantInter
f
## [1] 0.4769769
IntegracionCompuesta(limiteInferior = 1, limiteSuperior = 3, funcion = expression(x/(x^2+4)), cantInter
## [1] 0.4777546
PuntoMedioCompuesta(limiteInferior = 1, limiteSuperior = 3, funcion = expression(x/(x^2+4)), n = 8)
## [1] 0.4787515

```

```

# n = 1. Trapecio
# n = 2. Simpson
IntegracionCompuesta(limiteInferior = 3, limiteSuperior = 5, funcion = expression(1/sqrt(x^2-4)), cantI
g
## [1] 0.6054975
IntegracionCompuesta(limiteInferior = 3, limiteSuperior = 5, funcion = expression(1/sqrt(x^2-4)), cantI
## [1] 0.6043941
PuntoMedioCompuesta(limiteInferior = 3, limiteSuperior = 5, funcion = expression(1/sqrt(x^2-4)), n = 8)
## [1] 0.6029608

```

```

# n = 1. Trapecio
# n = 2. Simpson
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = ((3/8)*pi), funcion = expression(tan(x)), can
h
## [1] 0.9709263
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = ((3/8)*pi), funcion = expression(tan(x)), can
## [1] 0.9610554
PuntoMedioCompuesta(limiteInferior = 0, limiteSuperior = ((3/8)*pi), funcion = expression(tan(x)), n = 8
## [1] 0.9478684

```

Ejercicio 4

```

# n = 1. Trapecio
# n = 2. Simpson
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 1000, funcion = expression(x^(9.23-1)*exp(-x)).
4.1 a
## [1] 66173.21

```

```
gamma <- IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 1000, funcion = expression(x^(9.23-1.
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 1000, funcion = expression(0.43*exp(-0.43*x)*
```

4.1 b

```
## [1] 1
```

```
gamma <- IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 1000, funcion = expression(x^(9.23-1.
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 1000, funcion = expression(x * 0.43*exp(-0.43
```

4.1 c

```
## [1] 21.46512
```

4.1 d No me da

```
gamma <- IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 1000, funcion = expression(x^(9.23-1.
esperanza <- IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 20000, funcion = expression(x * (
IntegracionCompuesta(limiteInferior = 0, limiteSuperior = 10000, funcion = expression((x-esperanza)^2 *
```

```
## [1] 49.91888
```

```
mu <- 24
sigma <- 18
IntegracionCompuesta(limiteInferior = -10000, limiteSuperior = 10000, funcion = expression( (1/(sigma*s
```

4.2 a

```
## [1] 1
```

```
mu <- 24
sigma <- 18
IntegracionCompuesta(limiteInferior = -10000, limiteSuperior = 10000, funcion = expression(x * (1/(sigma
```

4.2 b

```
## [1] 24
```

```
mu <- 24
sigma <- 18
esperanza <- IntegracionCompuesta(limiteInferior = -10000, limiteSuperior = 10000, funcion = expression
IntegracionCompuesta(limiteInferior = -10000, limiteSuperior = 10000, funcion = expression((x-esperanza
```

4.2 c

```
## [1] 324
```