

Económicas, UBA. Actuario. Análisis Numérico.

Cuatrimestre 1, 2021. Primer Examen Parcial.

Reemplace este texto por su Apellido y Nombre, y su Numero de Registro

30/Abril/2021

Contents

1 Resolución de Ecuaciones: Newton-Raphson. (24 puntos)	2
1.1 Corregir algoritmo NR	2
1.2 Graficar función e identificar raíces para NR	2
1.3 Hallar raíces con NR	3
1.4 Iteraciones del algoritmo de NR	3
2 Resolución de Ecuaciones: Falsa Posición. (24 puntos)	5
2.1 Teoría	5
2.2 Iteraciones del algoritmo de <i>Regula Falsi</i>	5
2.3 Graficar función para RF y marcar raíces	5
3 Factorización de Matrices. (12 puntos)	7
3.1 Factorización de Cholesky	7
3.2 Factorización LU	7
4 Interpolación de Lagrange (40 Puntos)	8
4.1 Polinomio de Newton: $P_N(x)$	8
4.2 Interpolar con $P_N(x)$	8
4.3 Cubic Splines: $S_i(x)$	8
4.4 Interpolar con $S_i(x)$	8
4.5 Graficar	8
4.6 Comentar Resultados	9

Warning: package 'flextable' was built under R version 4.1.1

Warning: package 'tidyverse' was built under R version 4.1.1

Warning: package 'tibble' was built under R version 4.1.1

Warning: package 'tidyr' was built under R version 4.1.1

Warning: package 'readr' was built under R version 4.1.1

Warning: package 'purrr' was built under R version 4.1.1

Warning: package 'dplyr' was built under R version 4.1.1

Warning: package 'stringr' was built under R version 4.1.1

Warning: package 'forcats' was built under R version 4.1.1

1 Resolución de Ecuaciones: Newton-Raphson. (24 puntos)

Considere la siguiente ecuación: $2\sin(x) = e^{10/x}$.

1.1 Corregir algoritmo NR

CORREGIR el algoritmo “Newton-Rapshon” en el siguiente bloque de código.

COMENTAR los cambios que realizó (use “#” al final de cada línea modificada).

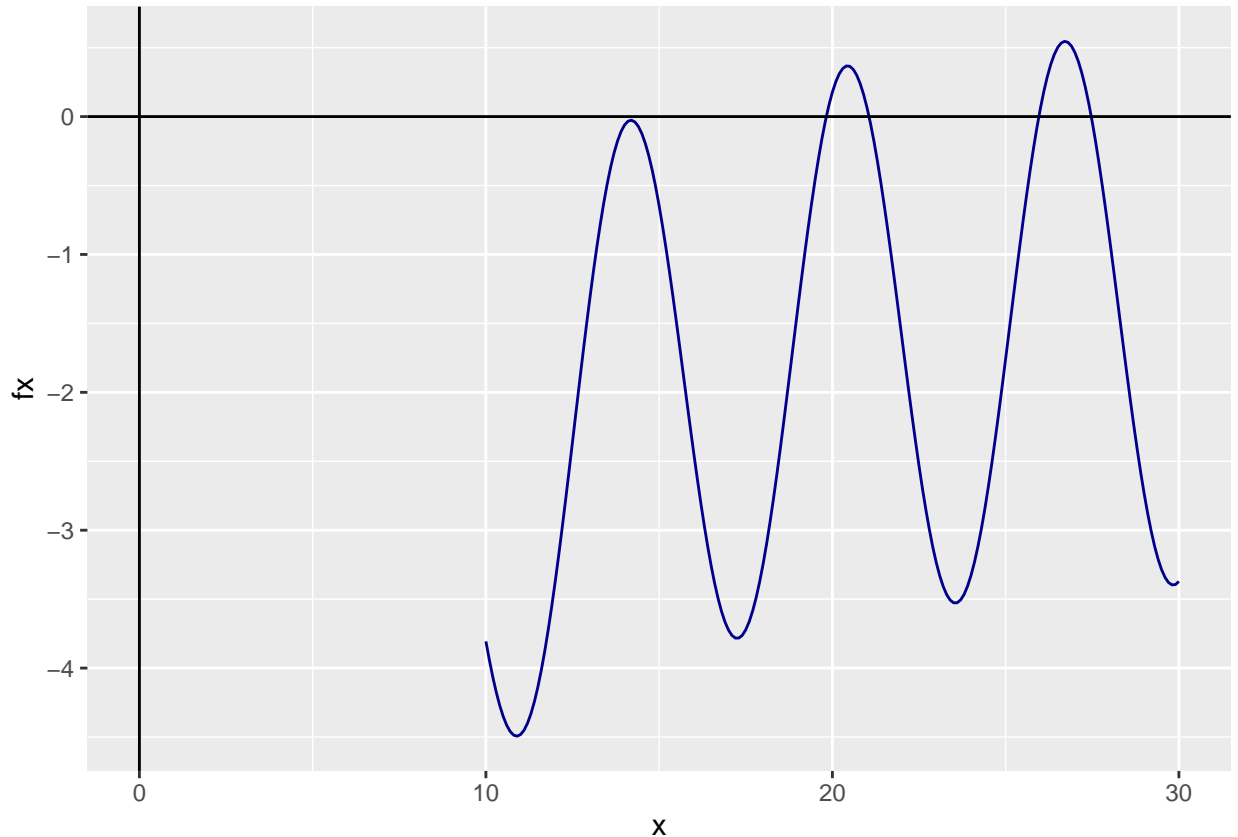
[Notar que, para el resto del ejercicio, no debe utilizar su propio algoritmo, sino que debe usar el algoritmo dado y corregido, sin agregar ninguna línea de código adicional.]

Respuesta:

```
# Edite las líneas que considere erróneas
NewtonRapshon <- function(f,df,p0,Tol,N){
  #Paso 1
  i = 1
  #Paso 2
  while (i <= N){
    #Paso 3
    p = p0 - f(p0)/df(p0) #df(p0)/f(p0)
    #Paso 4
    if (abs(p-p0) < Tol){return(p0)} #abs[p0-p]. Agregue los corchetes
    #Paso 5
    i = i + 1
    #Paso 6
    p0 = p
  }
  #Paso 7
  return(paste('El método falla luego de ', N, ' iteraciones')) #cambie n por N
}
```

1.2 Graficar función e identificar raíces para NR

Plantee la ecuación de la forma $f(x) = 0$ y grafique la función en el intervalo $[10; 30]$ de manera tal que pueda identificar todas las soluciones de la ecuación en el intervalo.



Respuesta:

1.3 Hallar raíces con NR

Utilizando el algoritmo del punto 1.1, halle todas las raíces identificadas en el punto 1.2.

Respuesta:

```
## [1] 19.8251
## [1] 21.05729
## [1] 25.95837
## [1] 27.47118
```

1.4 Iteraciones del algoritmo de NR

Tome el algoritmo del punto 1.1 (copie y pegue) y agregue las líneas de código que considere necesarias para poder visualizar (imprimir) cada iteración del algoritmo. Una vez editado el algoritmo, imprima 7 iteraciones del algoritmo iniciando en $x_0 = 17$. ¿A cuál de las raíces convergería el algoritmo en este caso?

Respuesta:

```
f <- function(x){
  return(2*sin(x)-exp(10/x))
}
NewtonRapshon <- function(f,df,p0,Tol,N){
  #Paso 1
  i = 1
  #Paso 2
```

```

while (i <= N){
  #Paso 3
  p = p0 - f(p0)/df(p0) #df(p0)/f(p0)
  #Paso 4
  if (abs(p-p0) < Tol){return(p0)} #abs[p0-p]. Agregue los corchetes
  #Paso 5
  i = i + 1
  #Paso 6
  p0 = p

  #Para que imprima 7 veces
  if (i <= 7) {
    print(p0)
  }
}
#Paso 7
return(paste('El método falla luego de ', N, ' iteraciones')) #cambie n por N
}

p0 <- 17
NewtonRapshon(f = f(x=p0), df = df(x=p0), p0 = p0, 0.0001, 100)

## [1] 9.369901
## [1] 7.690464
## [1] 9.484114
## [1] 7.702266
## [1] 9.535513
## [1] 7.698462
## [1] "El método falla luego de 100 iteraciones"

```

2 Resolución de Ecuaciones: Falsa Posición. (24 puntos)

Para este ejercicio, considere la función $h(x) = x^2 \times \cos(x)$.

2.1 Teoría

Explique las diferencias del método *Regula Falsi* respecto del método de *Secante*.

Respuesta (escriba a continuación): El método de la *posición falsa* genera aproximaciones de igual forma que el método de la *secante*, pero incluye una prueba para probar que las raíces se agrupan en iteraciones sucesivas.
Hallar raíces con RF {#raícesRF} Halle todas las raíces de $h(x)$ en el intervalo $[0, 3\pi]$ utilizando el método de Falsa Posición.

Respuesta:

[1] 1.570796

[1] 4.712389

[1] 7.853982

2.2 Iteraciones del algoritmo de *Regula Falsi*

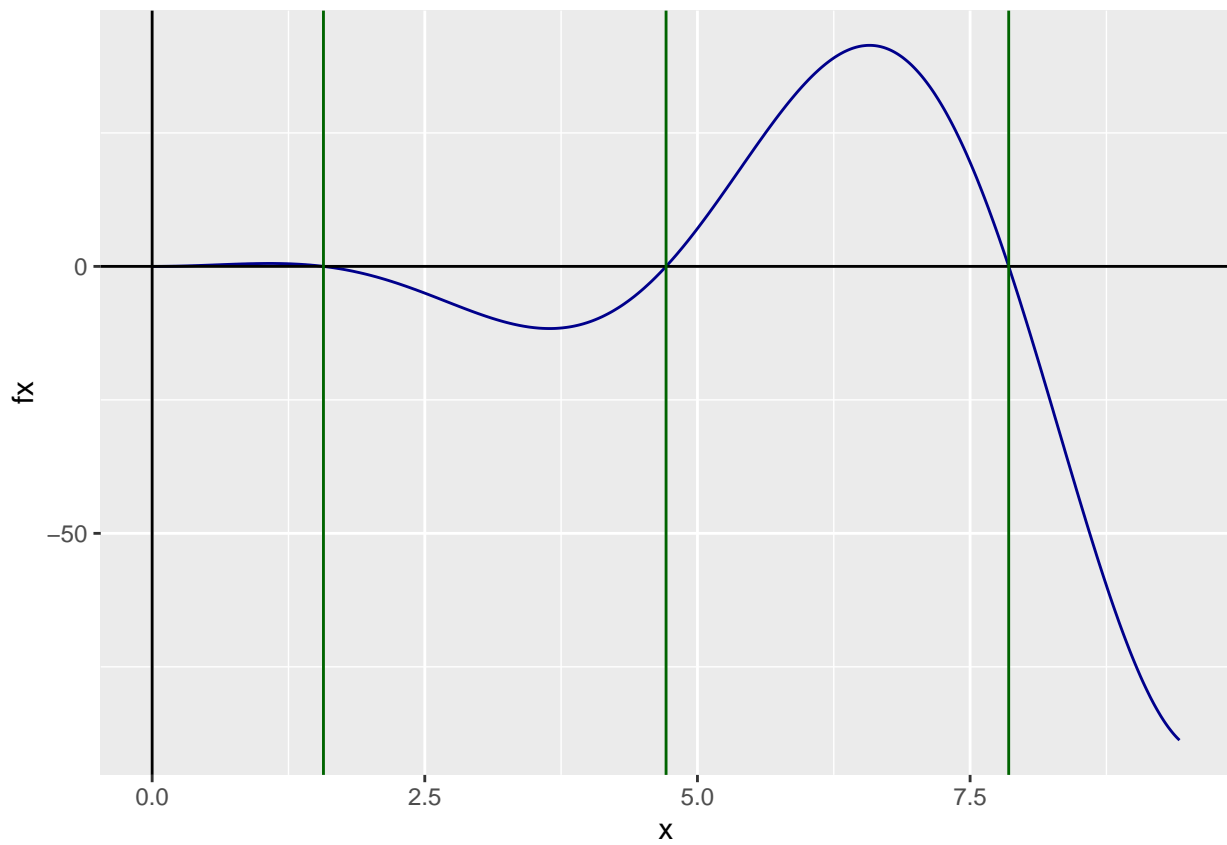
Considere el punto anterior. Realice 0 iteraciones del algoritmo de Falsa Posición, utilizando en $x_0 = 5$ y $x_1 = 6$. ¿A cuál de las raíces convergería el algoritmo en este caso?

Respuesta:

[1] 4.741887

2.3 Graficar función para RF y marcar raíces

Grafique la función $h(x)$ en el intervalo $[0; 3\pi]$, identifique todas las raíces halladas en el punto 2.2 y marque cada una con un color distinto en el gráfico (con un punto en el eje x o con una línea vertical).



Respuesta:

3 Factorización de Matrices. (12 puntos)

3.1 Factorización de Cholesky

Realice la factorización de Cholesky de la siguiente matriz:

$$A = \begin{bmatrix} 1 & 0.0481 & 0.0105 \\ 0.0481 & 1 & 0.0116 \\ 0.0105 & 0.0116 & 1 \end{bmatrix}$$

Respuesta:

```
##          [,1]          [,2]          [,3]
## [1,] 1.0000 0.00000000 0.00000000
## [2,] 0.0481 0.99884253 0.00000000
## [3,] 0.0105 0.01110781 0.9998832
```

3.2 Factorización LU

Realice la factorización de LU de la siguiente matriz:

$$B = \begin{bmatrix} 20 & 18 & 16 & 18 \\ 16 & 15 & 29 & 18 \\ 19 & 21 & 15 & 26 \\ 19 & 3 & 16 & 19 \end{bmatrix}$$

Respuesta:

```
## $L
##          [,1] [,2]          [,3] [,4]
## [1,] 1.00    0.0  0.000000    0
## [2,] 0.80    1.0  0.000000    0
## [3,] 0.95    6.5  1.000000    0
## [4,] 0.95 -23.5 -3.616114    1
##
## $U
##          [,1] [,2] [,3]          [,4]
## [1,] 20 18.0 16.0 18.00000
## [2,] 0 0.6 16.2 3.60000
## [3,] 0 0.0 -105.5 -14.50000
## [4,] 0 0.0 0.0 34.06635

##          [,1] [,2] [,3] [,4]
## [1,] 20 18 16 18
## [2,] 16 15 29 18
## [3,] 19 21 15 26
## [4,] 19 3 16 19
```

4 Interpolación de Lagrange (40 Puntos)

Considere la siguiente tabla de datos:

x	y
-2.0	0.0769
-1.5	0.1290
-1.0	0.2500
-0.5	0.5714
0.0	1.0000
0.5	0.5714
1.0	0.2500
1.5	0.1290
2.0	0.0769

4.1 Polinomio de Newton: $P_N(x)$

Escriba el Polinomio interpolante de Newton, $P_N(x)$, que pasa por todos los puntos dados

Respuesta:

4.2 Interpolar con $P_N(x)$

Calcule $P_N(1.25)$.

Respuesta:

4.3 Cubic Splines: $S_i(x)$

Escriba los trazadores cúbicos, $S_i(x); i = 1, \dots, n$ que pasan por todos los puntos dados. Indique claramente qué polinomio S_i se debe usar en cada intervalo de x .

Respuesta:

4.4 Interpolar con $S_i(x)$

Usando los trazadores cúbicos, interpole los datos para el valor $x = 1.25$

Respuesta:

4.5 Graficar

Grafique lo siguiente:

- Datos dados en la tabla mediante puntos (círculos rellenos o no).
- Línea continua de color azul con la función $P_N(x)$ para x en $[-2; 2]$.
- Línea continua de color rojo con los trazadores cúbicos para x en $[-2; 2]$.

Respuesta:

4.6 Comentar Resultados

A partir de lo hallado en el punto anterior, comente sobre la calidad de los métodos para realizar aproximaciones de la función entre los puntos dados.

Respuesta (escriba sus comentarios a continuación):