

Actividad | #1 | Análisis de Conceptos

Ingeniería en Desarrollo de Software



TUTOR: Miguel Angel Rodríguez Vega

ALUMNO: Edgar Enrique Cuamea Ochoa

FECHA: 06 de marzo del 2024

Contenido

Introducción.	3
Descripción.	4
Justificación.	5
Desarrollo.	6
Conclusión.	13

Introducción.

Utilizaremos Rstudio para la realización de distintos algoritmos para poder solucionar algunos problemas matemáticos utilizando operaciones menos complejas, por lo que utilizaremos este programa para realizar diferentes funciones tales como poder realizar representaciones de cantidades y variables, guardar valores fraccionados y realizar operaciones básicas como divisiones o multiplicaciones y obtener un resultado, así como poder modificar el como es que nos puede entregar el resultado, esto con el fin de poder utilizar los métodos numéricos para poder realizar aproximaciones de operaciones matemáticas complejas para poder solucionar un problema complejo de una forma más fácil utilizando este programa o incluso programas de manejo de datos como Excel ya que necesitaremos definir variables y asignarle valores así como la traficación de las funciones utilizando los diferentes métodos numéricos para encontrar el valor aproximado para resolver el problema matemático complejo utilizando algoritmos y funciones para encontrar el margen de error así como el valor aproximado de la incógnita X .

Descripcion.

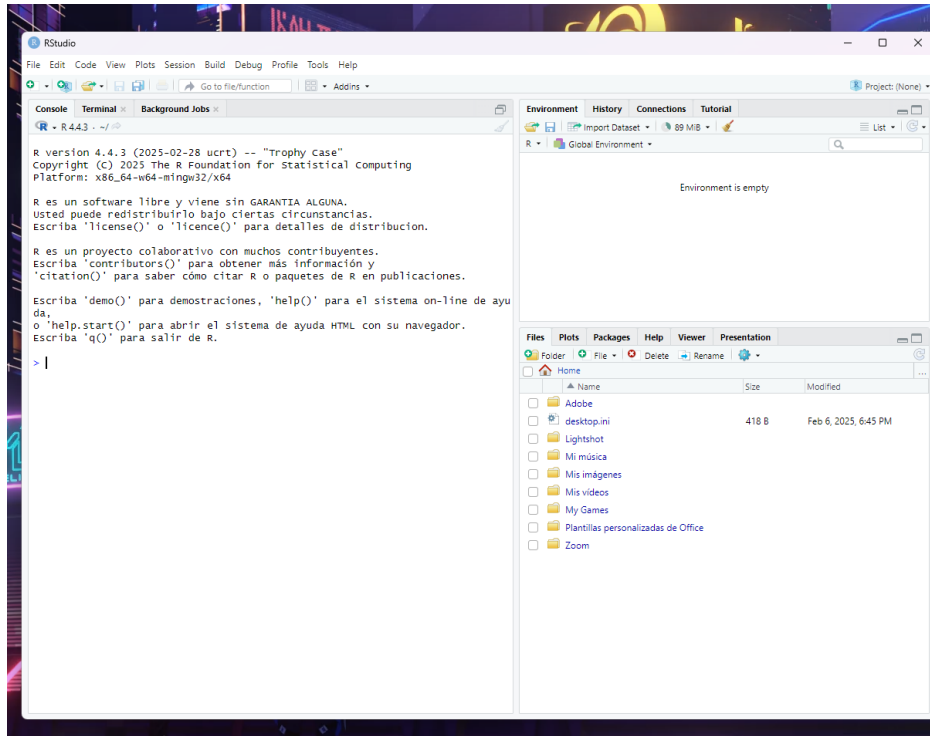
Tendremos en cuenta los diferentes métodos numéricos para poder encontrar el valor aproximado de una incógnita de un problema matemático complejo por lo que existen diferentes métodos que nos ayudaran a encontrar el valor aproximado, a través de estos métodos podremos utilizar diferentes formas de encontrar la incógnita utilizando diferentes funciones como tangentes o derivar funciones para optimizar el algoritmo y así poder encontrar el resultado aproximado de la forma más rápida posible sin tener que hacer tantas interacciones así como poder realizar operaciones más sencillas que nos tomaran mas interacciones para obtener el mismo resultado por lo que usaremos el programa de Rstudio para la realización de la actividad por lo que podremos optar por utilizar diferentes programas como Exel que nos permiten manipular datos para poder realizar graficas y tablas que nos llevaran al mismo resultados utilizando los métodos numéricos por lo que podremos elegir entre todos estos el que mas nos funcione para la realización de la operación así como la obtención de los resultados aproximados.

Justificación.

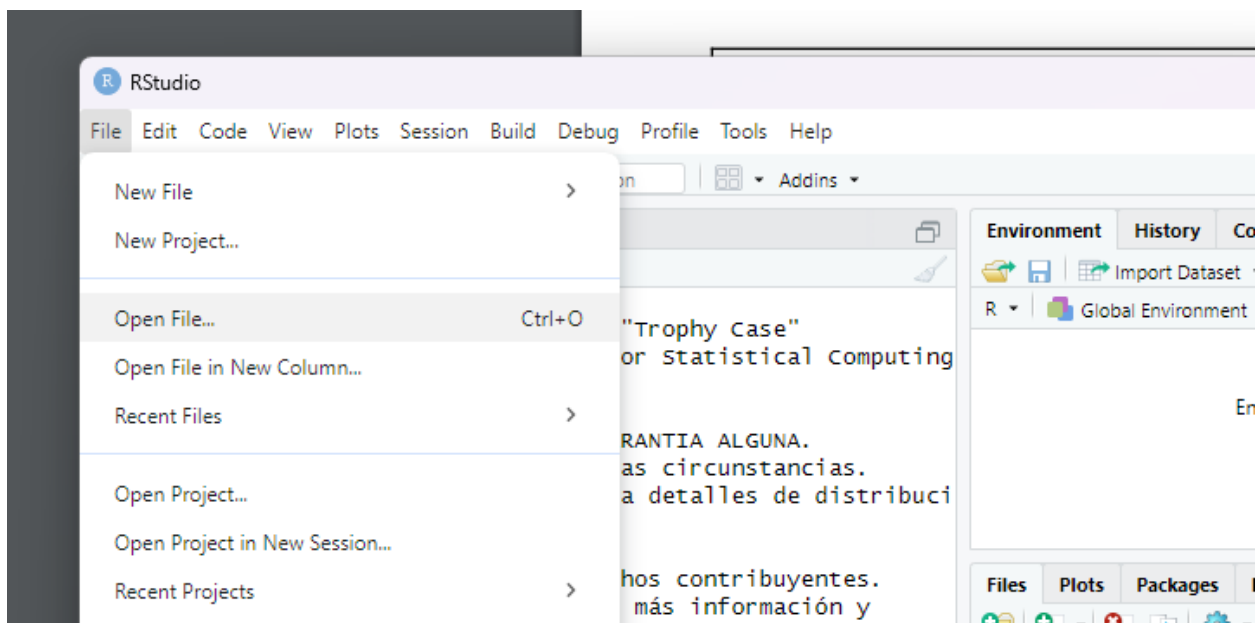
Aprenderemos a utilizar Rstudio para la realización de la actividad ya que es un programa para la manipulación de datos que nos funcionara para la realización de algoritmos complejos así como utilizar un lenguaje que esta especializado en el análisis estadístico, este programa también nos proporciona la visualización de datos ya que podremos crear graficas que nos ayudaran a saber qué tipo de método podremos utilizar para la resolución del problema y obtener el resultado aproximado, este programa contiene funciones matemáticas y numéricas que incluyen cálculos, integraciones, diferenciación, interpolación, resolución de ecuaciones diferenciales, entre otras, podremos crear simulaciones para así poder realizar análisis de errores, además cuenta con paquetes especializados para el uso de métodos numéricos para el análisis matemático para ecuaciones diferenciales y funciones por lo que lo hace un programa practico para la resolución de problemas matemáticos complejos con el uso de funciones e interacciones repetitivas para llegar al resultado aproximado del problema.

Desarrollo.

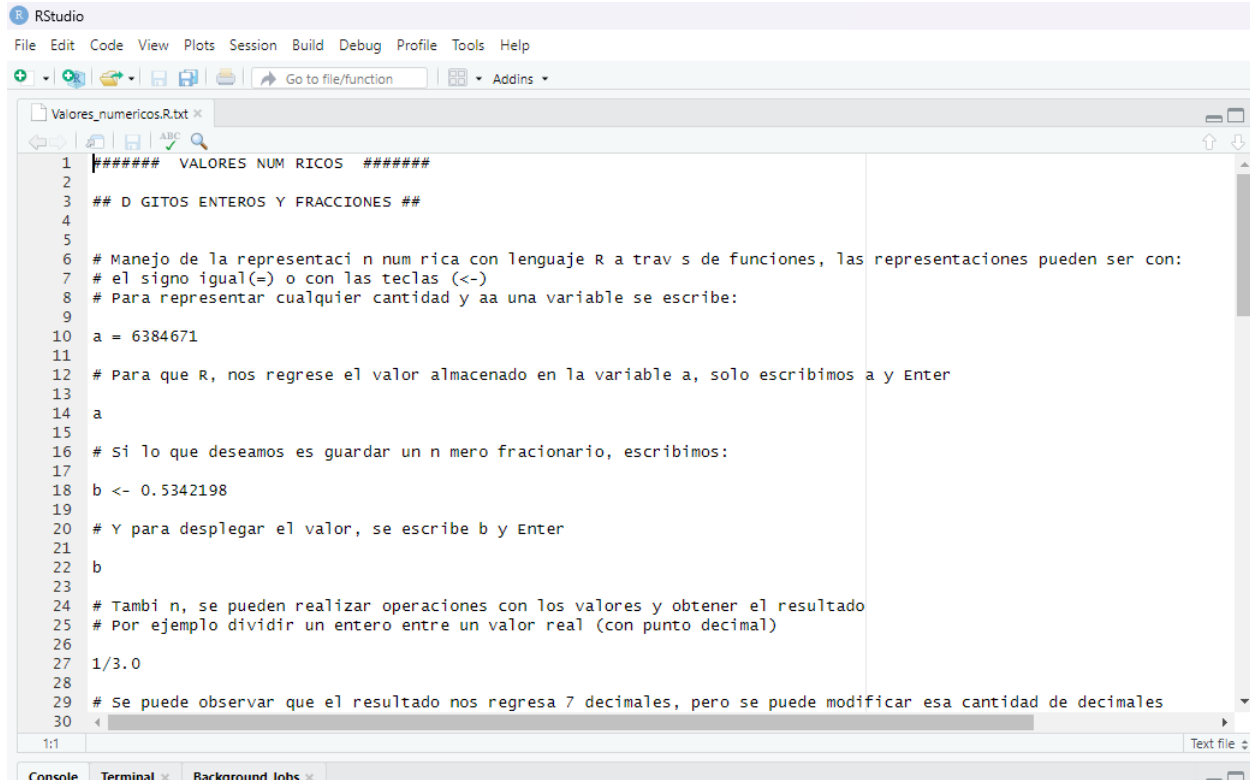
Descargamos desde la pagina oficial Rstudio, una vez descargado, abrimos el programa



Una vez abierto, descargamos el archivo de valores numericos desde el pdf de nuestra actividad o copiamos su contenido en un bloc de notas con el nombre de Valores_numericos.R



Esto hara que el archivo sea visible en Rstudio



```

1 ##### VALORES NUMERICOS #####
2
3 ## DIGITOS ENTEROS Y FRACCIONES ##
4
5
6 # Manejo de la representaci n num rica con lenguaje R a trav s de funciones, las representaciones pueden ser con:
7 # el signo igual(=) o con las teclas (<=)
8 # Para representar cualquier cantidad y aa una variable se escribe:
9
10 a = 6384671
11
12 # Para que R, nos regrese el valor almacenado en la variable a, solo escribimos a y Enter
13
14 a
15
16 # Si lo que deseamos es guardar un n mero fraccionario, escribimos:
17
18 b <- 0.5342198
19
20 # Y para desplegar el valor, se escribe b y Enter
21
22 b
23
24 # Tambi n, se pueden realizar operaciones con los valores y obtener el resultado
25 # Por ejemplo dividir un entero entre un valor real (con punto decimal)
26
27 1/3.0
28
29 # Se puede observar que el resultado nos regresa 7 decimales, pero se puede modificar esa cantidad de decimales
30

```

Ya que tenemos nuestro archivo, lo ejecutaremos por partes para realizar las operaciones por separado por lo tanto realizaremos lo siguiente:

```

> a = 6384671
> |

```

Con este comando podremos asignar un valor numerico a alguna variable, en este caso le asignamos el valor de 6384671 a la variable a, este lo podremos realizar con el signo (=) o utilizando (<=) para asignar valores

```

> a = 6384671
> a
[1] 6384671
> |

```

Escribiendo en la consola la letra de la variable y presionar enter podremos ver el valor

que contiene esta variable por lo que nos devuelve el resultado que le asignamos a la variable.

```
[1] 0.5342198
> b=0.5342198
> b
[1] 0.5342198
> |
```

Ademas de poder guardar en las variables numeros enteros enteros, podremos guardar numeros fraccionados, o aquellos que tiene decimales por lo que podremos realizarlo con el mismo proceso.

```
> 1/3.0
[1] 0.3333333
>
```

Podremos realizar operaciones matematicas como dividir 1 sobre 3.0 lo que nos da como resultado su valor numerico fraccionado de 0.3333333

```
> options(digits=3)
> 1/3.0
[1] 0.333
>
>
```

Con el comando de options(digits=3) podremos cambiar la cantidad de los decimales que nos arroja la operación, en ya que seleccionamos 3 decimales en vez de 7 decimales que estaban anteriormente por lo que podremos revertirlo utilizando otra vez el mismo comando pero cambiando el valor de los digitos para ver diferentes longitudes de decimales, ya que este comando se mantendra en todas las operaciones hasta que reiniciemos Rstudio

```
> round(54.2)
[1] 54
> |
```

Utilizando el comando round(x,n) podremos redondear los decimales donde x es el valor numerico y n el numero de decimales por lo que nos sirve para redondear valores con decimales


```
> round(97.5684197, 2)
[1] 97.57
> |
```

Al tener un valor con mas decimales que las que queremos que este nos muestre, al utilizar este comando nos devolvera el resultado con los decimales que le pedimos de formaredondeada por lo que 97.568 se convierte en 97.57 redondeando la operación.

```
> signif(27.384956102)
[1] 27.385
> signif(39.6429304521, 5)
[1] 39.643
> signif(61.378045912, 2)
[1] 61
> signif(316.6971243547, 3)
[1] 317
> |
```

Con el comando signif(x,n) podremos realizar la misma operación de redondeo donde x es el valor que ingresemos y n representa el numero de digitos que se mostraran en pantalla

```
> e <- exp(1)
> e
[1] 2.718282
> |
```

Con la expresion de e <- exp(1) asignamos un valor a la variable e en base a los logaritmos naturales que nos da como resultado 2.718282

```
> x = 0.005
> x0 = e ** (2*x)
> |
```

Podremos asignar valores con el simbolo de igual ademas de asignarle una funcion de e ** (2*x) a la variable de x0

```
> tex = "El valor de x0 es: "
> |
```

Incluso podremos asignarle una cadena de valores a la variable `tex` con las comillas para poder asignar texto dentro de esta variable

```
> cat(tex, x0)
El valor de x0 es: 1.01005
> |
```

Con la función `cat(tex, x0)` se obtiene la instrucción de `cat` que requiere mostrar la variable `tex` mas la función resuelta de `x0` por lo que nos muestra el mensaje completo

```
> x0 = 1
> x1 = x0 - pi * x0 + 1
> x1
[1] -1.141593
> cat("x0 =", x0, "\n", "x1 =", x1)
x0 = 1
x1 = -1.141593
> |
```

Asignamos valores a `x0` por lo que `x1` es igual a la variable `x0` menos π multiplicado por `x0` que es igual a 1 mas 1 por lo que al mostrar `x1` este nos da el resultado de -1.141593 ademas utilizamos la función de `cat` para resolver el problema que muestre `x0 =`, muestre el valor de `x0`, muestre el valor de `\n` por lo que si utilizamos `\n`, este cambia de renglon para mostrar los valores por separado ademas de mostrar el valor de `x1=` y `x1`, por lo que nos mostrara el mismo resultado de el valor de `x0` y `x1` por separado

```
> d = function(a,b,c) b^2-4*a*c
> d(2,2,1)
[1] -4
> |
```

Podremos realizar operaciones aritmeticas utilizando funciones donde $b^2-4*a*c$ y asignando el valor de `a,b,c` con diferentes numeros, este realizara la operación mostrando el resultado de la operación

```

g = function(x) sin(cos(x)*exp(-x/2))
plot(g, -8, -5,                                # Rango
      lwd = 1,                                # Grosor
      main = "Gráfico de g",                  # Título del gráfico
      col = "red",                            # Color de la línea
      xlab = "x",                             # Etiqueta de x
      ylab="g(x)",                           # Etiqueta de y
      axes = TRUE,                           # Ejes x,y visibles
      n = 1000)                              # Número de puntos

```

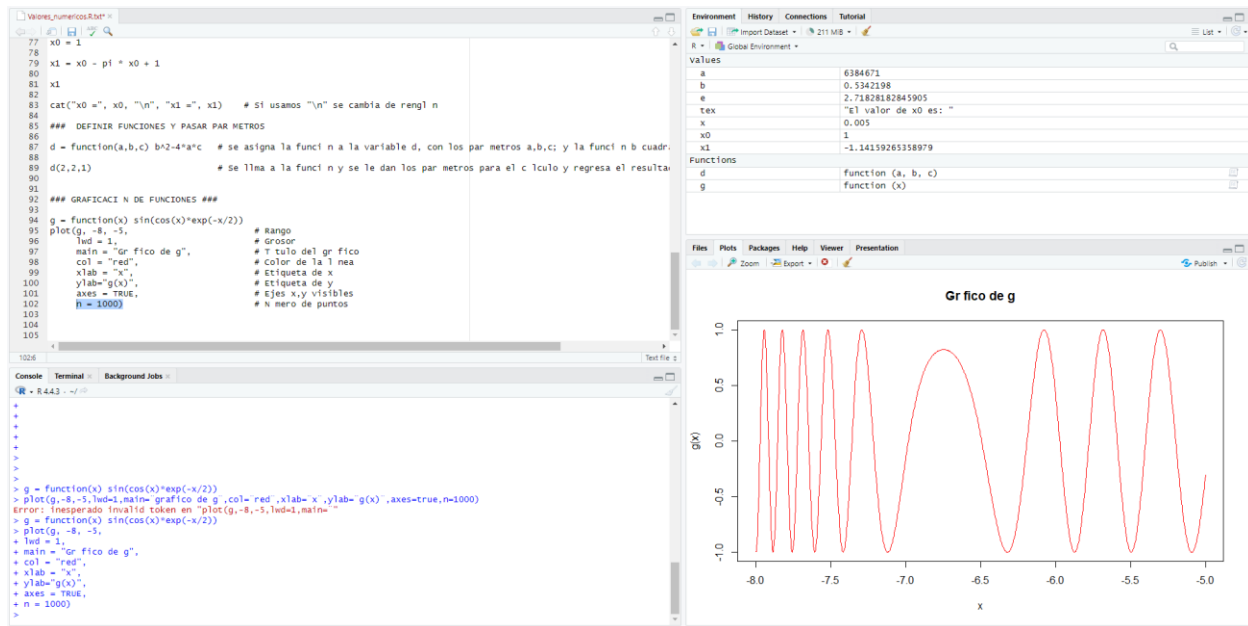
Con estas funciones podremos generar gráficos así como asignar colores, nombres para la tabla y etiquetas de los ejes X y Y así como el número de puntos que este tendrá

```

> g = function(x) sin(cos(x)*exp(-x/2))
> plot(g,-8,-5,lwd=1,main="gráfico de g",col="red",xlab="x",ylab="g(x)",axes=true,n=1000)
Error: inesperado invalid token en "plot(g,-8,-5,lwd=1,main="
> g = function(x) sin(cos(x)*exp(-x/2))
> plot(g, -8, -5,
+ lwd = 1,
+ main = "Gráfico de g",
+ col = "red",
+ xlab = "x",
+ ylab="g(x)",
+ axes = TRUE,
+ n = 1000)
>

```

Esta función es necesaria de ejecutarla por partes para que procece cada uno de los comandos ya que si lo ingresamos en una sola línea, este no funcionara por lo que una vez ejecutado el código, este mostrara una gráfica en la derecha con los valores que especificamos así como los valores y las variables de todas las operaciones realizadas arriba a la derecha



Por lo que entendiendo diferentes comandos para la utilización de este programa será necesario para la manipulación de datos así como poder generar gráficas y resolución de problemas matemáticos complejos de una forma más fácil.

Conclusion.

Aprendimos algunas operaciones básicas para la manipulación de datos así como poder realizar algunas operaciones como la asignación de datos a algunas variables, ya sean números enteros o fraccionados con decimales, realizamos algunas modificaciones para que nos muestre mas o menos decimales de las operaciones así como poder realizar operaciones con redondeos de decimales y mostrar datos con un numero de dígitos que especificamos en las operaciones, además de asignar expresiones de logaritmos que nos servirán al momento de utilizar los métodos numéricos mas adelante por lo que es importante tener esto en cuenta para poder asignar texto y el resultado de los logaritmos en un mismo renglón para mostrar el resultado, realizamos funciones donde ingresando diferentes datos numéricos, este realizara la operación matemática en base a los números que definimos en las variables además de tener la opción de poder realizar graficas de estas funciones logarítmicas así como la realización de estas operaciones matemáticas complejas para poder obtener una gráfica de manera rápida y sencilla solamente agregando valores de una manera sencilla de realizar.