Escuela de Ciencias Físicas y Matemáticas
Laboratorio de Simulación
Examen Parcial No.2
27/04/2020
Felipe Antonio Ixcamparic Choy

Resumen

En la toma de datos se puede buscar una forma de predecir como se comportará a futuro el fenómeno en estudio. Para ello existen distintos métodos matemáticos para ver si dichos datos se ajustan a alguno de ellos. A continuación se presentará el uso de uno de ellos, mejor conocido como Regresión lineal y, además otro para hallar raíces de una función dada conocido como el método de bisección aplicados a lenguajes de programación, específicamente el lenguaje C.

1. Objetivos

1.1. Generales

- Aprender el uso y aplicación de distintos métodos numéricos

1.2. Específicos

- Realizar un programa que dé el estime el comportamiento de las divisas entre 2 países.
- Encontrar la raíz de una función usando el método de bisección mediante un programa.

2. Planteamiento del problema

2.1. Problema 1 : Valor de la moneda

El comportamiento del dolar estadounidense ante el euro esta dado en la siguiente gráfica:

Cuadro 1: Tabla de año vs valor

Año	Euros
2006	1.003
2007	0.961
2008	0.928
2009	0.913
2010	0.880
2011	0.846
2012	0.807
2013	0.766
2014	0.734
2015	0.713
2016	0.692
2017	0.675
2018	0.656
2019	0.638

Asumiendo que el comportamiento siempre es el mismo. Se pide un programa que dé las siguientes soluciones :

1. Una gráfica que compare los valores tabulados y la recta que mejor aproxima el decrecimiento.

2. El año en el que el dolar no valdrá nada.

2.2. Problema 2: Hallar la raíz de una función

Se busca hallar la raíz de la siguiente función: $f(x) = 0.1x^2 + xln(x)$ mediante un programa que use el método de bisección.

3. Marco Teórico

3.1. Métodos numéricos

Para la realizar algunas operaciones, el caso analítico puede no ser la mejor forma de abordarlo, para ello existen los métodos numéricos.

"Son denominados así porque, usualmente, consisten en realizar una sucesión más o menos larga de operaciones numéricas (normalmente mediante la ayuda de un ordenador), al cabo de las cuales encontramos un valor numérico que, si bien no es la solución exacta del problema, se le parece mucho, es decir, aproxima la solución buscada con una precisión razonablemente buena."[1, p. 1]
Para abordar éstos problemas se utilizarán 2 métodos numéricos:

3.2. Mínimos cuadrados:

El método de mínimos cuadrados o regresión lineal consiste en ser una ”forma de análisis de regresión matemática utilizada para determinar la línea de mejor ajuste para un conjunto de datos, proporcionando una demostración visual de la relación entre los puntos de datos. Cada punto de datos representa la relación entre una variable independiente conocida y una variable dependiente”[2, p 1]
El método da una justificación para ajustar una linea entre los puntos, ya que su aplicación mas común es para un ajuste lineal de datos. Esto para que se minimice la suma de los cuadrados en los errores dados por los resultados con sus ecuaciones asociadas, para dar una predicción de como se podrían seguir comportando los datos.
Como objetivo de el ajuste lineal se tiene como objetivo ajustar a una función lineal de la forma

$$y = mx + b. \tag{1}$$

En donde **m** es llamada la y **b** el intersecto en y, o bien, el corte en y.
El método conlleva a que, teniendo un conjunto de datos ordenados xy. Se pueden obtener los 2 elementos (m y b) con las siguientes fórmulas.[3]

$$m = \frac{n \sum x_i y_i - (\sum_1^n x_i)(\sum_1^n y_i)}{n \sum_1^n x_i - (\sum_1^n x_i)^2} \tag{2}$$

$$b = \frac{\sum_1^n y_i - m \sum_1^n x_i}{n} \tag{3}$$

Además, cabe recalcar, que como toda medida científica se tiene error. Por lo que conlleva a que las fórmulas para el cálculo de los errores en ambos sea:

$$\Delta m = \frac{\sqrt{n}\varepsilon}{\sqrt{n \sum_1^n x_i^2 - (\sum_1^n x_i)^2}} \tag{4}$$

$$\Delta b = \frac{\varepsilon}{n} \tag{5}$$

Donde ϵ es el error de la variable x.

$$r = \frac{n \sum_1^n (x_i y_i) - \sum_1^n x_i \sum_1^n y_i}{\sqrt{n \sum_1^n x_i^2 - (\sum_1^n x_i)^2 (n \sum_1^n y_i^2 - (\sum_1^n y_i)^2)}} \tag{6}$$

Donde **r** es el coeficiente de correlación, el cual está comprendido entre -1 y 1. ”Si se acerca a +1, indica que es una correlación directa, si es -1 es una correlación inversa, y si se acerca a 0, quiere decir que no hay correlación o bien, es muy vaga”.[3]
En pocas palabras mientras mas se acerque a +-1, indicará que tan bien se ajusta.

3.3. Método de bisección:

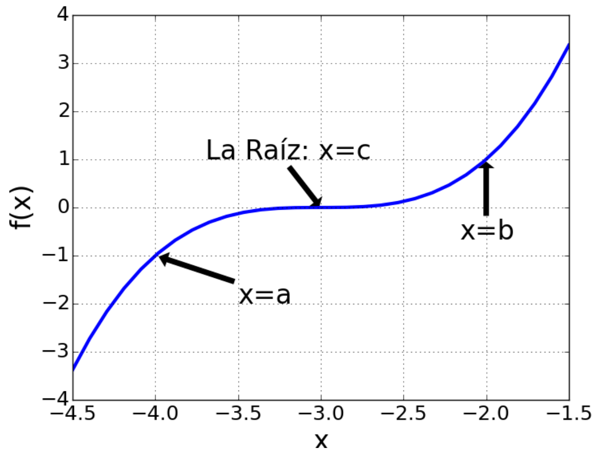
El método de bisección surge de plantear el problema de encontrar una raíz de una función de la forma

$$f(x) = 0 \tag{7}$$

"Desde el punto de vista geométrico, esto significa encontrar, en [a, b], un punto de corte de la gráfica de la

$$y = f(x) \tag{8}$$

con el eje de las abcisas."[1, p 4]



[4]

Funcionamiento en sí:

A grandes rasgos (y lo meramente necesario para éstos problemas) viene dado en los siguientes pasos.[1, p 6]

1. Definir dos cotas en donde se verá probable donde se halle una raíz en la función. (puede ser a y b)
2. Definir una media entre las 2 cota, llámesele h=(a+b)/2.
3. Observar que ocurre con el producto de la función valuada en ésta media y la cota inferior y observar si se comporta mayor o menor.
4. Para el caso menor a 0 se corre la cota mayor al h, y caso inverso si es mayor a 0.
5. Se repite éste proceso hasta llegar a una diferencia muy pequeña entre ambas cotas, o bien que el producto de un valor 0, lo cual indicaría que es la raíz exacta.

En un sentido matemático estrícto se anotaría como:

$$f(a) * f(h) < 0 \tag{9}$$

Se reemplaza c por h, pero si

$$f(a) * f(h) > 0 \tag{10}$$

Se reemplaza a por b. Se repite éste proceso como se indicó anteriormente.
Para el error se establece mediante el valor absoluto de las cotas enésimas.

$$\epsilon = |a_n - b_n| \tag{11}$$

3.4. Conceptos básicos de gnuplot

Es un programa de comandos con orientación gráfica en ordenador. Siendo originalmente creado para uso de estudiantes científicos, resulta ser uno de los mas grandes *software* de graficación mundial. Utilizado tanto por físicos y matemáticos, como por ingenieros.
Dada su importancia para su implementación en cuanto a gráficos, se explicarán algunos comandos importantes (y mas que todo relevantes) para los problemas planteados:

- plot[funcion]: El comando plot se encarga de graficar una función, anteriormente declarada en el entorno de gnuplot. Su sintaxis consta de primero anotar la función (la cual se almacena en su memoria) y seguidamente se puede graficar. El sistema por defecto lo realizará mediante una línea.
- replot[funcion]: Si ya se ha graficado una función antes, el comando replot se encarga de graficar otra función en la misma imagen generada por el programa. En pocas palabras, tener 2 gráficas en 1.[5, p 109]
- set labelx,y,etc "Título ": Se encarga de darle un nombre a algún eje de la gráfica o bien, cambiar el título de la misma. [5, p 25]
- set yrange o xrange[a:b]: Muestra el rango deseado de datos en la gráfica.[5, p 26]

3.4.1. Graficando archivos de texto

Si se tiene un archivo de texto con puntos ordenados x,y. Gnupolot es capaz de graficarlo mediante el comando anteriormente mencionado *plot*. La sintaxis, vendría siendo la misma que para una función delcarada, con la diferencia que ahora se indica el documento de texto a Gnuplot.
$plot'archivo_de_texto.txt'$ se encargaría de graficar mediante una gráfica de dispersión (por defecto).

3.4.2. Leyendo instrucciones para Gnuplot

Gnuplot cuenta también con un comando llamado **load** el cual se encarga de leer instrucciones indicadas específicamente al mismo.
Se puede crear uno con la extensión .gp una serie de comandos separados por líneas de por medio para que gnuplor al utlizar el comando **load** los ejecute. [5, p 84]

4. Metodología

4.1. Problema 1

Para abordar éste problema, se realizará mediante C un programa encargado de tomar los datos y ajustarlos mediante regresión lineal, finalizando por exportar la función con la

pendiente y el intersecto en ella, además de un documento de texto en donde se impriman los datos ya mencionados.
Para su ejecución se creará un script en bash, encargado de compilar, ejecutar y graficar los datos obtenidos de nuestro programa anterior mediante gnuplot. Por lo tanto se dividirá en 2 secciones siendo C la primera y el script de bash la segunda.

4.1.1. Programa en C

Para éste programa se tendrá contemplado utlizar 3 funciones: Principal o 'main', el método de mínimos cuadrados, exportador a documento de texto. Además de varabables globales para la pendiente ,las respectivas sumatorias, los errores de las pendientes y el error de x (para éste caso se seteara en 0.1 ya no hay un "$errror$" medido.

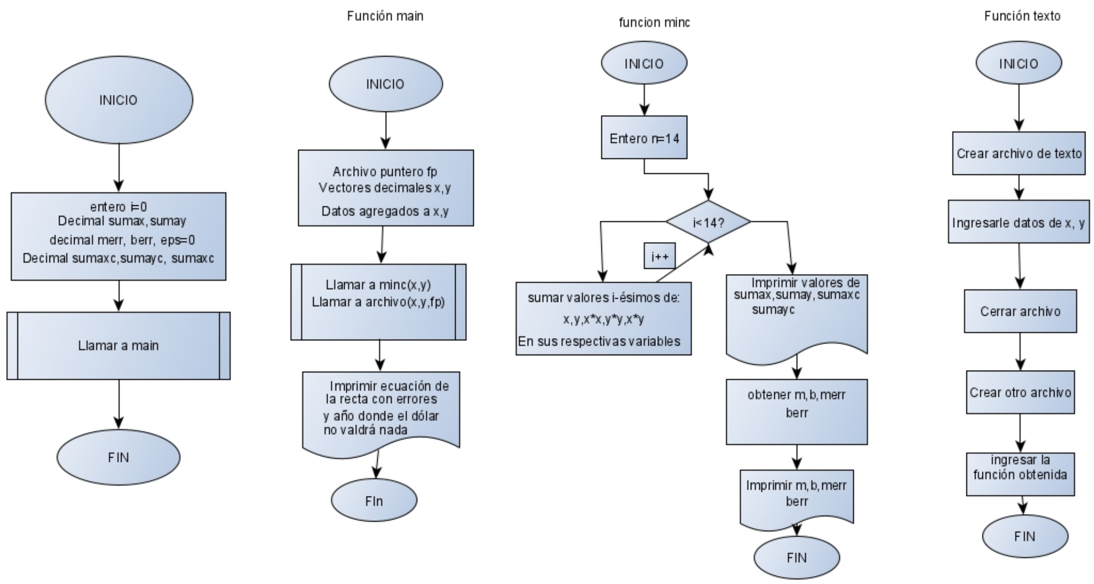
El main se encargará de almacenar en 2 vectores distintos los datos de la tabla (numero), siendo el vector x quien almacene los años y 'y' los valores. Además acá se imprimirá el resultado hacia el usuario visible en consola.

La segunda función se encargará de realizar cada sumatoria de x,y,xy,x^2 y y^2 respectivamente. Acto seguido se utilizará la ecuación (2) para la pendiente y la ec.(3) para el y intersecto. La forma de realizar cada sumatoria será mediante un for loop sumando cada componente del vector x e y según sea necesario para cada uno. Se ha de decir que los resultados se almacenarán en variables globales llamadas m y b (pendiente e intersecto).

Para la última función se crea un puntero el cual hará la labor de crear 2 distintos archivos de texto en donde el primero almacenará los datos de x e y (la tabla) y el segundo almacenará la función obtenida mediante el método de mínimos cuadrados. Se ha de recalcar que la función que se almacenará ocupará un número mayor de cifras significativas para que el ajuste tenga la mejor precisión posible, ya que, en el main se desplegará con menores cifras para mera visualización del usuario.

Diagrama de Flujo:

Para una mejor visualización antes de realizar un pseudocódigo (o código en sí) se esboza el siguiente diagrama de flujo:



Pseudocódigo:

A continuación se muestra un pseudocódigo para tener una idéa de como iría la estructura de la programación de éste programa.

1. Incio
2. Se declaran variables globales m=0,b=0,i=0,n=14, sumax,y,xc,yc,xy. Tambien se declara merr,berr, eps=0.1. Seguido de la funcion para mínimos cuadrados (minc) y para almacenar en archivo (archivo)
3. Se inicia la función main: Se declaran vectores locales x e y almacenando años y valores respectivamente.Además de un puntero fp.
4. Se llama a la función minc y archivo ambas valuadas en 'x' e 'y'.
5. Se imprime la función resultante y merr,berr.
6. Finalizar función.
7. Para función minc: Declarar variables locales x,y,xc,yc,xy.
8. Inicio for, si $i < n$ entonces realizar suma de x[i],y[i],$x[i]^2$,$y[i]^2$,$x[i]y[i]$ respectivamente y almacenarlo en cada una de ellas.
9. En caso contrario, imprimir el valor de sumax,sumay,$sumax^2$,$sumay^2$,$sumaxy$.
10. Obtener pendiente y el interesecto por la fórmula dada, luego almacenarlo en m y b.

11. Imprimir m,b.
12. obtener merr y berr
13. Finalizar función.
14. Para la función texto: Usar el puntero y crear un archivo para redactar llamado "Datos.txt".
15. Inicio for, i=0, si $i < n$ entonces imprimir x[i] , y[i] separados por un espacio, luego dejar una línea.
16. En caso contrario terminar de redactar en el archivo.
17. Abrir el puntero y crear otro archivo para redactar llamado "texto.gp".
18. Anotar la función obtenida con mínimos cuadrados.
19. Terminar de redactar.
20. Finalizar función

4.1.2. Script de Bash

El script de bash hará el favor de compilar, ejecutar y darle la instrucción de gnuplot de que grafique ambos archivos de datos en uno. Siendo mas estricto el procedimiento sería:

1. Iniciar bash
2. Compilar mediante gcc -lm -o (lm por que se usa $< math.h >$ y -o para renombrarlo como ./Cuadrados
3. Ejectuar el archivo compilado ./Cuadrados
4. Hacerle "$echo$" al archivo de instrucciones (el texto.gp) tales como $setxrange[2006 : 2019]$ (Por los años que se dan) $setxlabel, settitle, entreotros$
5. Indicar como última instrucción $plotf(x)$ y $replot"datos.txt"$ para tener en una misma gráfica.
6. Indicarle a gnuplot que lea el archivo con las instrucciones mediante el comando gnuplot -p "$archivo.gp$"
7. Eliminar ambos archivos de datos para no confundir al usuario. mediante -rm "datos.txtz rm "texto.gp".

4.2. Problema 2

Para éste problema nuevamente se tomará la estructura de un programa en C, el cual será compilado y luego graficando la función mediante un script en bash.
Se usará el método de bisección en el programa en C, dejando en consola el resultado de la raíz y el script se encargará realizar la gráfica de la función.

4.2.1. Programa en C

Para éste programa se tiene contemplado utilizar 3 funciones: La principal o 'main' , la función indicada y el método de bisección.

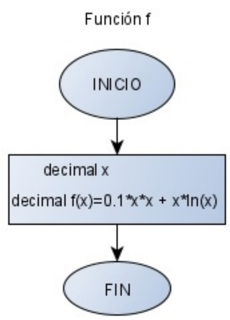
El main se encargará de crear variables que almacenarán los límites, ya que la función tiene una raíz definida en el intervalo [0:1] se eligen éstos 2 como las cotas de ella para que allí se ejecute el método. Luego llamará a la función de bisección, finalmente imprimirá el valor de la raíz de la función en ese intervalo.

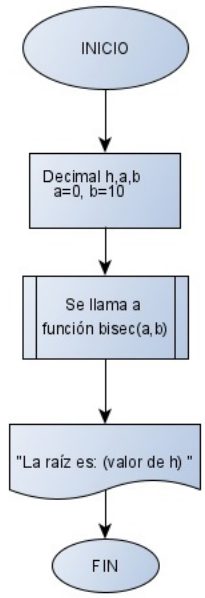
La función ocupará a $f(x) = 0.1x^2 + xln(x)$.

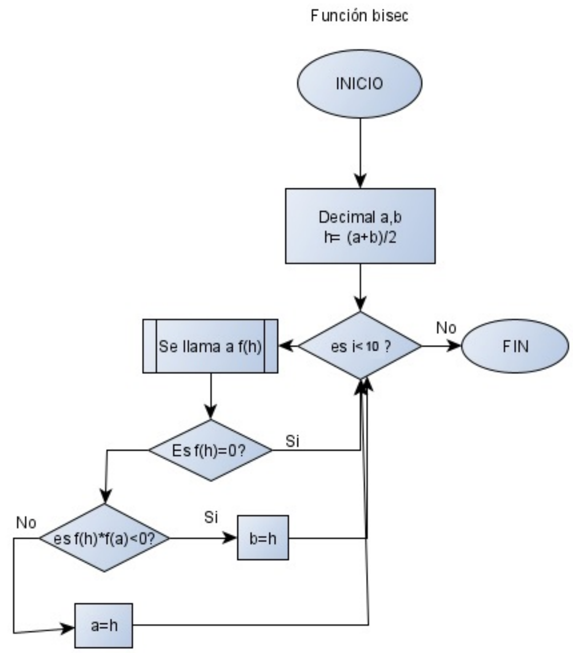
La función que realizará el método de bisección iniciará estableciendo el valor de h como el valor "medio"de las dos cotas. Luego con un for loop de iteraciones arbitarias, en éste caso serán establecidas como 10, se revisarán los 3 posibles casos que supone el método de bisección : Si la función valuada en h es igual a 0, entonces el valor de h es la raíz y repite el ciclo hasta llegar a 10. Si la función valuada en h multiplicada con ella valuada en la cota inferior es menor que cero ($f(a) * f(h) < 0$) entonces se corre h a la cota superior y se repite el cículo. Por último si $f * (a) * f(h) > 0$ se desplzará h a la cota inferior y se repite el cíclo. Finalmente se devuelve el valor almacenado en h tras las 10 iteraciones.

Diagrama de Flujo:

Para una mejor visualización antes de realizar el pseudocódigo se realiza el siguiente diagrama de flujo:







Pseudocódigo:

A continuación se verá el respectivo pseudocódigo correspondiente al programa en C.

1. Incio
2. Se declara la variable global h y err. Seguido de la función para biseccion (bisec) y la función de la función (valga la redundancia).
3. Se inicia la función main: Se declaran las variables locales a=0 y b=1.
4. Leer n
5. Se llama a la función de bisección y se valúa en las 2 cotas (bisec(a,b)).
6. Imprimir el valor de h y el de err.
7. Para función f, crear variable local x, y retornar $f(x) = 0.1x^2 + x * ln(x)$
8. Fin de la función.

9. Para función bisec: Inicio for loop de 0 a n, h=(a+b)/2.
10. Llamar a f(a,b). Si f(h)=0 entonces regresar valor de h.
11. Si $f(x) * f(a) < 0$ entonces b=h
12. En caso contrario a=h
13. Fin del for.
14. Regresar valor de h.
15. error = valor absoluto (b-a)/n*100
16. Fin de la función

4.2.2. Script de bash

El script de bash se encargará de compilar, ejecutar y realizar la gráfica de la función de la siguiente manera:

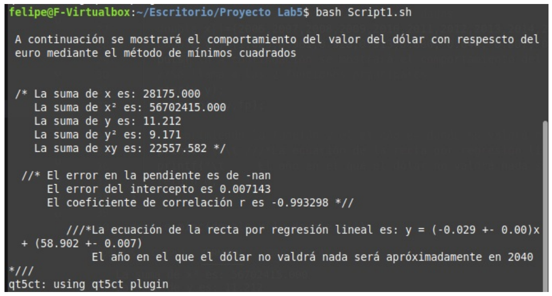
1. Compilar el programa mediante gcc -lm -o donde -lm es porque se usa la librería $< math.h >$, y -o para renombrarlo
2. Ejecutar el programa
3. Mediante "printf"se le envian instrucciones a un nuevo documento llamado bistec.gp tales como $setxlabel, settitle, etc$ para definir las etiquetas, y arreglar el rango de la raíz.
4. Se finalizan las instrucciones con la de plot f(x).
5. Se da la instruccion de ejecutar gnuplot y utilizar el comando load, simplificado como gnuplot -l "bistec.gp". De modo que realizará la gráfica de la función.

5. Resultados

5.1. Problema 1

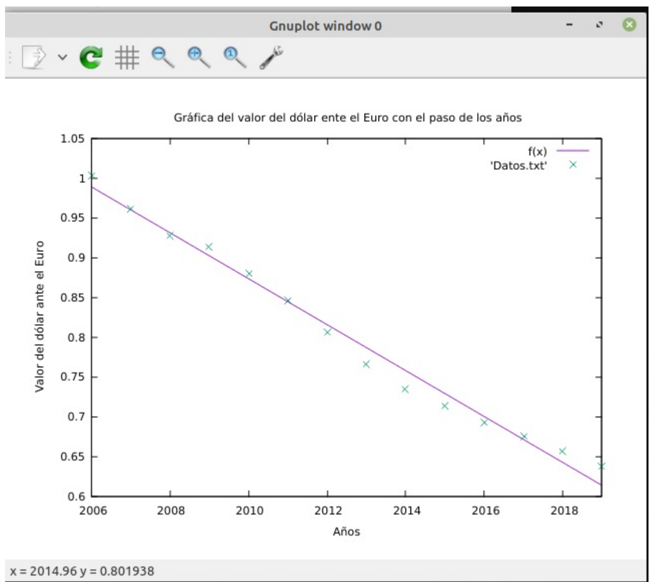
El código resultante de éste problema se puede acceder en el siguiente link: `https://github.com/felipechoy1/Lab2Simu/tree/master/Examen%20Paracial%20%232`

Tras ejecutar el problema, se obtienen los siguientes resultados en la siguiente imagen:



De donde es claro que la gráfica por el ajuste de mínimos cuadrados otorga la función $y = mx + b$ con y Además de un coeficiente de correlación $r = -0.9932$

En donde la gráfica en donde se comparan los datos y la función queda de la siguiente manera:

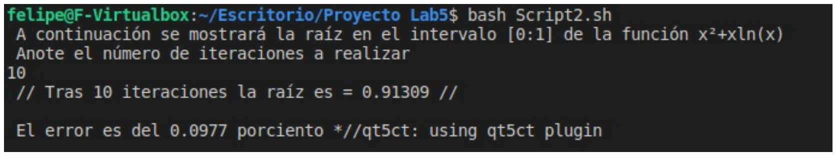


Ésto consecuentemente trae como resultado que el dólar no valdrá nada con repecto al euro en el año **2040**. Y además, notar que los puntos se acomplan en buena medida a la regresión lineal indicada. Con un coeficiente muy cercano a 1.

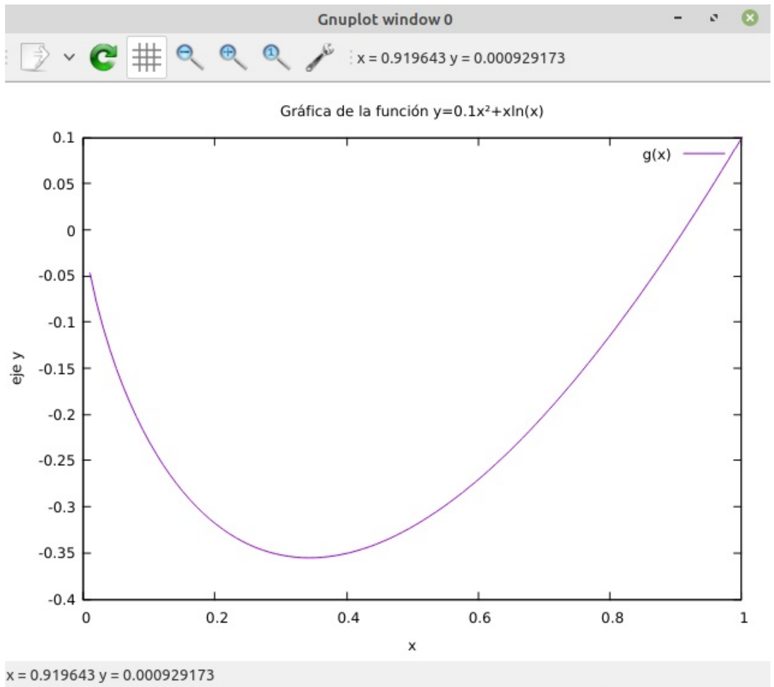
5.2. Problema 2

Tras crear el programa, se puede acceder al repositorio mediante el siguiente link: `https://github.com/felipechoy1/Lab2Simu/tree/master/Examen%20Paracial%20%232`

Tras ejecutar el segundo programa, se obtiene el siguiente resultado (ver imagen).



Entonces, en ese intervalo, según el método nos dará en **x=0.91309**. Con un error del **0.0977 %** Observando la gráfica de la función:



De donde, el corte en y es cercano a **x=0.912766**. Por lo tanto, la diferencia sería cercana a **0.000324**

6. Conclusiones

- El ajuste por mínimos cuadrados es funcional para predecir el comportamiento de la divisa del dólar en contra del Euro
- El dólar probablemente no valga nada con respecto del Euro en el año 2040
- El método de bisección se ajusta de buena manera con funciones que tengan 1 raíz.
- Se debe usar un número considerable de iteraciones para tener un resultado aceptable con el método de bisección.

Referencias

[1]anónimo, «Métodos Numéricos», (2019).

[2]W. Kenton, «Least Squares Method Definition», (2019).

[3]anónimo, «Regresión Lineal», (2010-2019).

[4]J. Ruiz, «unknown», (2019).

[5]T. Williams y V. a. Kelley Collin, *Gnuplot 5.2 An interactive plotting program*, *ver*.5.2 (United States, 2004-2019).