Series de potencias Taylor. Función Ln(x)

Yoselin Armas Ramos y Bianca E. Kennedy Giménez

GAII

Técnicas Experimentales. 1er *curso.* 2do *semestre*

Facultad de Matemáticas

Universidad de La Laguna

La Laguna, 15 de mayo de 2014

Índice general

Índice de figuras

Resumen

La asignatura de Técnicas Experimentales nos ha aportado varios conocimientos sobre lo que era para nosotros un nuevo lenguaje de programación. Estos conocimientos adquiridos serán empleados en la investigación de la Serie de Taylor aplicada a la función $ln(x)$. En el siguiente informe se puede observar el contenido del trabajo elaborado: historia, series de Taylor y Mclaurin, aplicación matemática y empleo de algoritmos para realizar varios experimentos que nos llevarán a determinadas conclusiones.

Capítulo 1

Motivación y objetivos

El objetivo de esta práctica es demostrar los conocimientos adquiridos en Latex, Beamer y Python en el transcurso de la asignatura de Técnicas Experimentales. Aplicaremos los programas ya mencionados en la realización de un informe sobre la función logaritmo neperiano y su desarrollo de Taylor. Además implementaremos un programa en Python que nos ayudará a realizar varios experimentos para respaldar nuestras afirmaciones sobre el tema planteado.

1.1. Uso de Python,Beamer y Latex

- LaTeX : Utilizaremos este programa en la realización del informe que presentaremos sobre $f(x) = Ln(x)$
- Beamer : Recurriremos a la creación de diapositivas para orientarnos en la exposición oral del trabajo.
- Python : Crearemos un programa en lenguaje interpretado Python que utilizará el desarrollo de Taylor para aproximar la función Ln(x). Así como calcularemos el tiempo que tarda en realizar dicha aproximación.

Capítulo 2

Fundamentos teóricos

2.1. Introducción

El desarrollo de Taylor calcula la aproximación del valor de una función centrada en un punto, así como la estimación del error. En este caso, investigaremos la función logartimo Neperiano de "x". Veremos la aproximación y el error aplicando la serie de Taylor, y las variaciones de los resultados dependiendo de las constantes que fijemos. Además, nos introduciremos en el tema hablando de la historia del desarrollo de Taylor y del Logaritmo Neperiano. Para ello utilizaremos tal y como hemos mencionado anteriormente,el lenguaje de programación Python elaborando programas que respalden las hipótesis dadas y nos aporte el resultdo esperado.

2.2. Historia

El filósofo griego Zenón fue uno de los primero en considera el problema de la suma de una serie infinita para obtener un resultado finito, tras varios estudios la consideró imposible, como resultado surgió la paradoja de Zenón. Más tarde, Aristóteles propuso una resolución filosófica a la paradoja.Su contenido matemático no fue resuelto hasta que lo retomaron Demócrito y Arquímedes a través del método de agotamiento de Arquímedes, que se basa en que un número infinito podría expresarse finalmente mediante un resultado finito.

En el siglo XIV, Madhava de Sangamagrama dio los primeros ejemplos de la utilización de la serie de Taylor y otros métodos relacionados; aunque no queda constancia de sus estudios, los escritos posteriores de matemáticos de la India sugieren que encontró algunos casos especiales de la serie de Taylor,como por ejemplo las funciones trigonométricas. Las series de Taylor tuvo gran relevancia en los estudios que realizaba la famosa escuela de Kerala de astronomía y matemáticas.

En el siglo XVII, James Gregory trabajó en esta área y publicó varias series de Taylor centradas en el punto cero. Sin embargo, no fue hasta el siglo XVIII cuando se presentó de manera formal el Desarrollo de Taylor que proporcionaba una solución finita para cualquier tipo de función. Esta aportación fue dada en 1715 por el matemático británico Brook Taylor (1685-1731).

$$\theta = \tan \theta - (1/3) \tan^3 \theta + (1/5) \tan^5 \theta - \cdots ,$$

[1]

Figura 2.1: Brook Taylor

El teorema de Taylor da estimaciones cuantitativas sobre el error en la aproximación de una función. Cualquier número finito de términos iniciales de la serie de Taylor de una función se llama polinomio de Taylor. La serie de Taylor de una función es el límite de los polinomios de Taylor de esa función, siempre que el límite existe. Una función no puede ser igual a su serie de Taylor, aunque su serie de Taylor converge en cada punto. Una función que es igual a su serie de Taylor en un intervalo abierto se conoce como una función analítica. Es importante mencionar que si aplicamos la Serie de Taylor en el punto "0", la serie se llamaría Desarrollo de Maclaurin expresada de la siguiente manera:

$$f(x) = f(0) + f^1(0)x + \frac{f^2(0)}{2!}x^2 + \frac{f^3(0)}{3!}x^3 + ... + \frac{f^n(0)}{n!}x^n$$

Donde:

- "x" es el punto.
- "0" es el centro.
- "n" es el grado.

[1]Gregory redescubrió un teorema originalmente formulado por el matemático indio Madhava de Sangamagrama, la serie del arcotangente

2.3. Serie de Taylor y Logaritmo Neperiano

2.3.1. Series de potencias: Taylor

En matemáticas, una serie de Taylor es una representación de una función como una infinita suma de términos que se calculan a partir de las derivadas de la función para un determinado valor de la variable. Para analizar el comportamiento de una funcion "f" en las proximadades de un punto 'a',podemos recurrir a la aproximación local cerca de dicho punto. De esta manera,podríamos sacar conclusiones sobre el comportamiento de la función en el punto 'a'. Los resultados que se obtengan serán tanto más precisos cuanto mayor sea la aproximación que se maneje cerca del punto en cuestión.

2.3.2. Logaritmo Neperiano

Estudiando los fenómenos de creciemiento y decrecimiento en la naturaleza, se observó que con frecuencia aprecían potencias de un número irracional al que se llamo "e", cuyo valor aproximado es:

$$e \approx 2,71828182845904523536028747135 27.$$

Para estudiar estos fenómenos se aplicaban los logaritmos y sus propiedades, en concreto el logaritmo en base "e", también llamado "logaritmo neperiano".

$$ln(x) = log_e(x)$$

2.4. Aplicación matemática del Teorema de Taylor y uso del Logaritmo neperiano

Sea f una función sufcientemente regular y x la variable, podemos aproximar la función, para x cerca de a, mediante polinomios denominados polinomios de Taylor cuya expresión es la siguiente:

$$P_{(n,a)} = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f^{(3)}(a)}{3!}(x-a)^3 + \cdots \frac{f^{(n)}(a)}{n!}(x-a)^n \,.$$

Donde la derivada de orden cero de la función es definida como la propia función, "$n!$" es el factorial de "n" y $f^{(n)}.(a)$ denota la n-ésima derivada de f para el valor a. Además,es importante mencionar que si a=0, la serie se denomina Serie de Maclaurin.

2.4.1. Aplicación Series de Taylor en f(x)=ln(x)

Se aplica el desarrollo de Taylor de manera general, donde:

$$n = grado$$

$$a = valor$$

$$f^t(a) = \frac{b}{a^t},$$

$$1 < t < n$$

Sea f(x)=ln(x) :

1. Se calcula la imagen: f(a) = ln(a)
2. Se calcula la primera derivada y su imagen: $f'(a) = \frac{1}{a}$
3. Se calcula la segunda derivada y su imagen: $f''(a) = \frac{-1}{a^2}$
4. Se calcula la tercera derivada y su imagen: $f'''(a) = \frac{2}{a^3}$
5. Calculamos la enésima derivada y su imagen: $f^n(a) = \frac{(n-1).(-b)}{a^{(n)}}$.

Entonces,

$$P_{(n,a)} = ln(a) + \frac{\frac{1}{a}}{1!}(x-a) + \frac{\frac{-1}{a^2}}{2!}(x-a)^2 + \frac{\frac{2}{a^3}}{3!}(x-a)^3 + \cdots \frac{\frac{(n-1).(-b)}{a^{(n)}}}{n!}(x-a)^n \,.$$

Capítulo 3

Procedimiento experimental

3.1. Descripción de los experimentos

Tras elaborar un programa en Python capaz de calcular la aproximación de la función ln(x) mediante la serie de Taylor y el tiempo que tarda la máquina en realizar esa aproximación, hemos realizado varios experimentos fijando el punto,el centro o el grado. Además,también elaboramos un programa que nos calcula el error de dicha aproximación. Los experimentos realizados son los siguientes:

1. Fijamos el grado de la Serie de Taylor(n=10),el punto en el que deseamos que se aplique (x=4) y variamos el centro.
2. Fijamos el grado de la Serie de Taylor (n=8), el centro (c=6) y variamos el punto en el que deseamos que se aplique.
3. Fijamos el centro(c=1) , el punto en el que deseamos que se aplique la aproximación (x=2) y variamos el grado.

3.2. Descripción del material

Todos los experimentos realizados se han llevado a cabo en un ordenador con las siguientes características:

```
- Sistema operativo : Linux, Ubuntu.
- Procesador: Intel(R) Core(TM) i3
- CPU :  M 350  @ 2.27GHz
```

3.3. Resultados obtenidos

3.3.1. Función real del Logartimo Neperiano

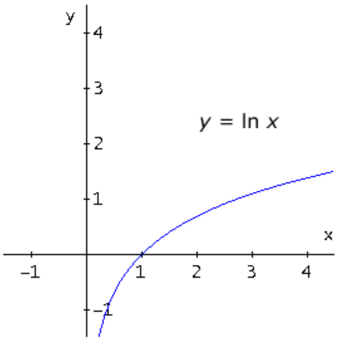

Figura 3.1: Función Logaritmo Neperiano

3.3.2. Variación del centro

Fijando n=10 y x=4

Si el valor de c= 2	1,33878210119487
Si el valor de c= 3	1,38629396788690
Si el valor de c= 5	1,38629436340045
Si el valor de c= 30	1,48572417754861
Si el valor de c= 60	1,74292118557045

3.3.3. Variación del punto

Fijando n=8 y c=6

Si el valor de x= 8	2,07943719692636
Si el valor de x= 7	1,94591013946629
Si el valor de x= 6	1,79175946922805
Si el valor de x = 5	1,60943792540920
Si el valor de x= 4	1,38630243959407

3.3.4. Variación del grado

Fijando x=2 y c=1

Si el valor de n= 5	0,783333333333333
Si el valor de n= 15	0,725371850371850
Si el valor de n= 25	0,712747499542829
Si el valor de n= 200	0,690653430481824
Si el valor de n= 800	0,692522571184642

3.3.5. Tiempo

Tomamos los resultados de la tabla anterior y calculamos el tiempo que tarda el ordenador en obtener los resultados.

Si el valor de n= 5	9,53674316406e-07
Si el valor de n= 15	9,53674316406e-07
Si el valor de n= 25	9,53674316406e-07
Si el valor de n= 200	1,19209289551e-06
Si el valor de n= 800	1,90734863281e-06

3.4. Análisis de los resultados

- Variación del centro: Variando el centro, fijando el grado (n=10) y el punto (x=4), podemos observar que cuanto más se aleja centro del punto menos acertado es el resultado.
- Variación del punto: Variando el punto, y fijando el centro (c=6) y el grado (n=8), observamos que si el punto es igual al centro el valor de la Serie de Taylor es no conduce a ningún error, ya que al ser iguales su diferencia es cero y nos da la imagen en ese punto o centro del logaritmo neperiano. Además, vemos que si incrementamos la diferencia entre ambos puntos el resultado se aleja más del auténtico valor.
- Variación del grado: Al variar el grado y fijando el punto (x=2) y el centro (c=2), apreciamos que cuanto mayor es el grado más disminuye el error, por lo cual, si aunmentamos el valor del grado más precisos son los resultados.
- Tiempo: en cuanto al tiempo, el ordenador realiza los procesos más lento dependiendo del grado, ya que la ecuación es más larga.

Capítulo 4

Conclusiones

Mediante la realización del trabajo hemos contrastado la información con programas informáticos (Python), por lo cual, queda demostrado la eficacia frente a otros métodos, ya sea por la facilidad que aporta al hacer investigaciones o por su preción de cálculo. Con esta herramienta de trabajo hemos concluido que el Desarrollo de Taylor aplicado a la función logaritmo neperiano de "x", fijando dos varibles y variando una, afecta, sobretodo, a la proximidad de la Serie de Taylor en f(x)=ln(x) y el valor del Logaritmo Neperiano en ese punto. Los resultados de la parte experimental, respaldados por la parte teórica en la que se demuestra como hallar el polinomio de Taylor del ln(x) mediante un sumartorio general o un sumatorio solo aplicable al logaritmo neperiano, nos indican que el error será minimo cuanto mayor sea el grado o nulo si el punto y el centro a tratar son iguales; también, hemos podido observar el efecto contrario al variar el centro. Además, hemos empleado los procesadores de texto LaTeX y Beamer, lo que nos ha llevado a ampliar nuestros conocimientos en esta materia.

Apéndice A

Apéndice 1

A.1. Algoritmo

Fichero .py

```
from sympy import *
import sys

def fac(grado):
  if grado == 0:
    return 1
  else:
    return grado * fac(grado-1)

def taylor(grado, punto, centro):
  c = Symbol('c')
  funcion = ln(c)
  suma = funcion.evalf(subs={c:centro})
  for i in range(1,grado+1):
    deriv = diff(funcion, c)
    termino = (deriv.evalf(subs={c:centro})/fac(i))*((punto-centro)**i)
    funcion = deriv
    suma += termino
  return suma

grado = int(raw_input('Introduzca el grado: '))
punto = float(raw_input('Introduzca el punto: '))
centro = float(raw_input('Introduzca el centro: '))
suma = taylor(grado, punto, centro)
print 'El polinomio de Taylor es: ', suma
```

Bibliografía

[1] http://www.buscabiografias.com/bios/biografia/verDetalle/2155/Brook Taylor.

[2] http://ordenador.wingwit.com/Programacion/python-programming/93912.html#.U2_W3vl5N1Y.

[3] http://fr.wikiversity.org.

[4] http://ordenador.wingwit.com/Programacion/python-programming/93912.html#.U2_XGPl5N1Y.

[5] http://es.scribd.com/doc/8557124/Series-de-Taylor-con-Python.

[6] http://www.slideshare.net/hermesx10/expansin-polinomial-en-series-de-taylor.

[7] http://recursostic.educacion.es/descartes/web/materiales_didacticos/Desarrollo_serie_taylor/Desarrollo_en_serie_

[8] Juan de Burgos. *Cálculo infinitesimal de una variable*. McGraw-Hill, Edificio Valrealty, 1ªPlanta.Basauri,17. Madrid., 2004.

[9] Michael Spivak. *Calculus:Cálculo infinitesimal*. Editorail Reverté,S.A, 1998.

[10] ACM LaTeX Style. http://www.acm.org/publications/latex_style/.