Integración: Simpson

$f(x) = x^2 \, cos \; x, \;\; x \in [1, 3]$

Adrián R. Mendióroz Morales
Roberto C. Palenzuela Criado

Grupo 2

Técnicas Experimentales. 1^{er} *curso.* 2^{do} *semestre*

Lenguajes y Sistemas Informáticos

Facultad de Matemáticas

Universidad de La Laguna

La Laguna, 13 de mayo de 2013

Índice general

1. Motivación y objetivos **1**
- 1.1. Objetivo principal 1

2. Fundamentos teóricos **2**
- 2.1. Integración Numérica. 2
- 2.2. Regla de Simpson 2
- 2.3. Regla de Simpson compuesta 5

3. Procedimiento experimental **6**
- 3.1. Descripción de los experimentos 6
  - 3.1.1. Representación gráfica de la función. 6
  - 3.1.2. Cálculo del valor exacto de la integral definida. 7
  - 3.1.3. Comparación gráfica. 8
  - 3.1.4. Aproximación por la regla de Simpson. 9
  - 3.1.5. Aproximación por la regla de Simpson compuesta. 9
- 3.2. Descripción del material 9
- 3.3. Resultados obtenidos 10
- 3.4. Análisis de los resultados 10

4. Conclusiones **11**

A. Ficheros Python **13**
- A.1. Grafica de la función 13
- A.2. Gráfica comparativa 14
- A.3. Modulo Simpson 15
- A.4. Aproximacion y error de la regla de Simpson 16
- A.5. Aproximacion y error de la regla de Simpson 16

Bibliografía **17**

Índice de figuras

2.1. Descripción gráfica de la regla de Simpson 3

3.1. Representación gráfica de la función a integrar. 7
3.2. Funcion vs. Polinomio interpolador 8

Índice de cuadros

3.1. Aproximación y error de la regla de Simpson 10
3.2. Aproximación y error de la regla de Simpson compuesta 10

Capítulo 1

Motivación y objetivos

Como finalidad de la asignatura de Técnicas Experimentales del Grado en Matemáticas, se debe saber aplicar conocimientos en el área de las Matemáticas de una forma profesional y poseer las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas.

1.1. Objetivo principal

Los matemáticos suelen encontrarse con el problema de integrar funciones que no están definidas de forma explícita. Se pueden utilizar métodos gráficos, pero los métodos numéricos son mucho más precisos.

El objetivo principal de este proyecto es investigar el método numérico "La Regla de Simpson" y aplicarlo sobre la integral de la función $f(x) = x^2\ cos\ x$, en el intervalo $[1, 3]$.

Capítulo 2

Fundamentos teóricos

La necesidad de aproximar numéricamente el valor de una integral surge por dos motivos fundamentalmente:

- la dificultad o imposibilidad en el cálculo de una primitiva,
- la función a integrar sólo se conoce por una tabla de valores.

2.1. Integración Numérica.

La integración numérica constituye una amplia gama de algoritmos para calcular el valor numérico de una integral definida y, por extensión, el término se usa a veces para describir algoritmos numéricos para resolver ecuaciones diferenciales. El problema básico considerado por la integración numérica es calcular una solución aproximada a la integral definida:

$$\int_a^b f(x)\ \mathrm{d}x$$

Los métodos más comunes de integración numérica son:

- La regla del Trapecio.
- La regla de Simpson.

2.2. Regla de Simpson

Una forma de obtener una aproximación adecuada de una integral es usar polinomios de grado superior para unir los puntos y aproximar la función real.

El método de Simpson, a diferencia de la Regla del Trapecio, intenta no recurrir a un mayor número de subdivisiones; se trata de ajustar una curva de orden superior en lugar de una línea recta como en la Regla del Trapecio.

La metodología será la siguiente: Sea una función $f(x)$, si $f(a) \neq f(b)$ entonces existe un punto intermedio $f(c)$ por el que se puede ajustar una parábola que una $f(a)$ y $f(b)$.

En la figura 2.1 se muestra la representación gráfica de una función real $f(x)$ y el polinomio de grado dos $P_2(x)$, que aproxima dicha función en un intervalo determinado. La regla de Simpson no es más que la fórmula resultante de tomar integrales bajo este polinomio.

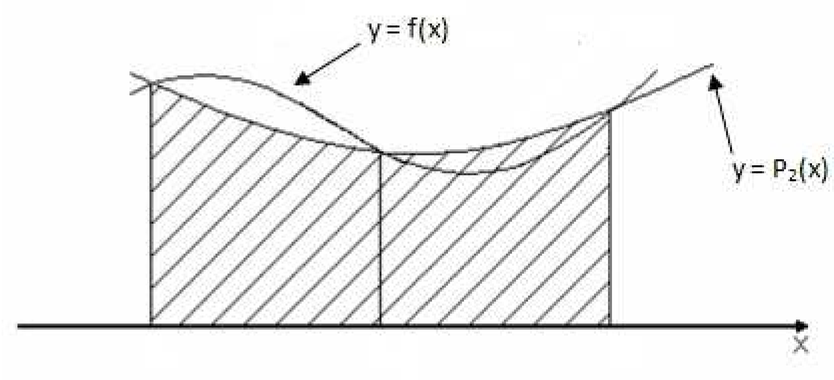

Figura 2.1: Descripción gráfica de la regla de Simpson

Utilizando una interpolación polinomial de segundo orden que aproxima a la función integrando $f(x)$ entre los puntos a y b, se desea aproximar la integral:

$$I = \int_a^b f(x)\mathrm{d}x$$

La interpolación polinomial que se busca es un polinomio de Lagrange de orden 2, $P_2(x)$, tal que, siendo $c = \frac{a+b}{2}$

$$P_2(x) = \frac{(x-c)(x-b)}{(a-c)(a-b)}f(a) + \frac{(x-a)(x-b)}{c-a)(c-b)}f(c) + \frac{(x-a)(x-c)}{(b-a)(b-c)}f(b)$$

Sustituyendo en la integral que se quiere calcular, se obtiene:

$$I = \int_a^b f(x)\mathrm{d}x \approx \int_a^b P_2(x)\mathrm{d}x$$

$$\int_a^b \left[\frac{(x-c)(x-b)}{(a-c)(a-b)}f(a) + \frac{(x-a)(x-b)}{(c-a)(c-b)}f(c) + \frac{(x-a)(x-c)}{(b-a)(b-c)}f(b)\right]\mathrm{d}x =$$

$$f(a)\int_a^b \frac{(x-c)(x-b)}{(a-c)(a-b)}\mathrm{d}x + f(c)\int_a^b \frac{(x-a)(x-b)}{(c-a)(c-b)}\mathrm{d}x + f(b)\int_a^b \frac{(x-a)(x-c)}{(b-a)(b-c)}\mathrm{d}x$$

Se toma: $h = \frac{b-a}{2}, \quad c = \frac{a+b}{2}$
Se despeja b de forma que:

$$h = \frac{b-a}{2} \Rightarrow b = a + 2h$$

Se sustituye la expresión $(a-c)(a-b)$ de la siguiente forma:

$$h = \frac{b-a}{2} \Rightarrow a-b = -2h$$

$$(a-b)(a-c) = -2h(a-c)$$

$$(a-b)(a-c) = -2h(b-2h-c)$$

$$(a-b)(a-c) = -2h\left(b-2h-\frac{a+b}{2}\right)$$

$$(a-b)(a-c) = -2h\left(\frac{b-a}{2}-2h\right)$$

$$(a-b)(a-c) = -2h(h-2h)$$

$$(a-b)(a-c) = -h^2$$

Obteniéndose así que:

$$\int_a^b \frac{(x-c)(x-b)}{(a-c)(a-b)}\mathrm{d}x = \frac{1}{2h^2}\int_a^{a+2h}(x-c)(x-b)\mathrm{d}x$$

Realizando un cambio de variable tal que $u = x-a$:

$$(x-c) = x-a+a-c = u+a-c$$

$$(x-c) = u+a-\frac{a+b}{2}$$

$$(x-c) = u+\frac{a-b}{2}$$

$$(x-c) = u-h$$

De la misma forma:

$$(x-b) = x-a+a-b = u+a-b$$

$$(x-b) = u-2\frac{b-a}{2}$$

$$(x-b) = u-2h$$

De lo que se obtiene que:

$$\int_a^b \frac{(x-c)(x-b)}{(a-c)(a-b)}\mathrm{d}x = \frac{1}{h^2}\int_a^{a+2h}(x-c)(x-b)\mathrm{d}x = \frac{1}{2h^2}\int_0^{2h}(u-h)(u-2h)\mathrm{d}u = \frac{h}{3}$$

Análogamente se deduce que:

$$\int_a^b \frac{(x-a)(x-b)}{(c-a)(c-b)}\mathrm{d}x = \frac{4h}{3} \quad ; \quad \int_a^b \frac{(x-a)(x-c)}{(b-a)(b-c)}\mathrm{d}x = \frac{h}{3}$$

Por tanto, se tiene que:

$$I = \int_a^b f(x)\mathrm{d}x \approx \frac{h}{3}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right]$$

Recordando que la expresión $h = \frac{b-a}{2}$, la expresión anterior se puede expresar como:

$$\int_a^b f(x)\mathrm{d}x \approx \frac{b-a}{6}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right] \tag{2.1}$$

A la ecuacion 2.1 se le conoce como "La regla de Simpson".
El término de error asociado es

$$-\frac{1}{90}(b-a)^5 f^{(4)}(\xi),$$

para algún $\xi \in (a, b)$.
Se observa que el error es proporcional a la cuarta derivada, de lo que se sigue que el método de Simpson obtiene soluciones exactas para ecuaciones de tercer grado o inferior.

2.3. Regla de Simpson compuesta

En casos donde la regla de Simpson devuelva un error elevado debido a la excesiva amplitud del intervalo $[a, b]$, se utiliza una variación de la regla de Simpson conocida como regla de Simpson compuesta, que consiste en dividir el intervalo $[a, b]$ en un número par "n" de subintervalos iguales de longitud $h = \frac{b-a}{n}$, entonces:

$$\int_a^b f(x)\mathrm{d}x = \int_a^{a+h} f(x)\mathrm{d}x + \int_{a+h}^{a+2h} f(x)\mathrm{d}x + \ldots + \int_{a+(n-1)h}^{b} f(x)\mathrm{d}x$$

Aplicando la Regla de Simpson a cada subintervalo ($x_i = a + ih; i = 0, 1, 2, \ldots, n$) se sigue que, para cada $[x_{j-1}, x_{j+1}]$ donde $j = 1, 3, 5, \ldots, n-1$, se obtiene que:

$$\int_{x_{j-1}}^{x_{j+1}} f(x)\mathrm{d}x = \frac{x_{j+1} - x_{j-1}}{3}\left[f(x_{j-1}) + 4f(x_j) + f(x_{j+1})\right]$$

Con lo que, la suma de las integrales de todos los subintervalos da como resultado:

$$\int_a^b f(x)\mathrm{d}x \approx \frac{h}{3}\left[f(x_0) + 2\sum_{j=1}^{\frac{n}{2}-1} f(x_{2j}) + 4\sum_{j=1}^{\frac{n}{2}} f(x_{2j-1}) + f(x_n)\right] \tag{2.2}$$

La ecuación 2.2 se conoce como "la regla de Simpson compuesta".
El término de error para esta fórmula es

$$-\frac{1}{180}(b-a)h^4 f^{(4)}(\xi)$$

para algún $\xi \in (a, b)$.

Capítulo 3

Procedimiento experimental

En este apartado se explicará el procedimiento que se ha seguido para obtener, tanto el valor exacto, como el valor aproximado a través de la regla de Simpson, de la integral definida:

$$\int_1^3 x^2 \; cos \; x \; \mathrm{d}x$$

A continuación se procederá a la comparación de ambos resultados. A modo de curiosidad se aplicará la regla de Simpson compuesta para observar la mejora en la aproximación.

3.1. Descripción de los experimentos

Los experimentos realizados son los siguientes:

- Representación gráfica de la función.
- Cálculo del valor exacto de la integral definida.
- Comparación grafica.
- Aproximación por la Regla de Simpson.
- Aproximación por la Regla de Simpson compuesta.

3.1.1. Representación gráfica de la función.

Se implementa en Python un programa que representa la gráfica de la función

$$f(x) = x^2 \; cos \; x \tag{3.1}$$

como se puede ver en el apéndice A.1, obteniendo como resultado la figura 3.1.

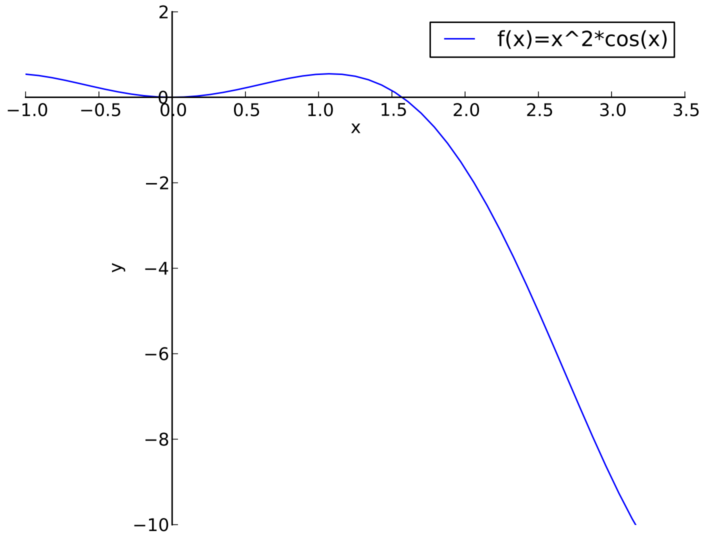

Figura 3.1: Representación gráfica de la función a integrar.

3.1.2. Cálculo del valor exacto de la integral definida.

A continuación se detallarán los pasos seguidos para obtener el valor exacto de la integral definida de la función 3.1 en el intervalo acotado $[1,3]$, utilizando el segundo teorema fundamental del cálculo integral (o Regla de Barrow).

Se quiere calcular:

$$\int_1^3 x^2 \ cos \ x \ \mathrm{d}x$$

Primero se resuelve la integral indefinida aplicando el método de integración por partes ($\int u\mathrm{d}v = uv - \int v\mathrm{d}u$) tal que:

$$\int x^2 \ cos \ x \ \mathrm{d}x =$$

$$u = x^2, \quad \mathrm{d}u = 2x \ \mathrm{d}x$$

$$\mathrm{d}v = cos \ x \ \mathrm{d}x, \quad v = sen \ x$$

Con lo que se obtiene:

$$\int x^2 \ cos \ x \ \mathrm{d}x = x^2 sen \ x - \int 2x sen \ x \ \mathrm{d}x =$$

Empleando de nuevo la integración por partes:

$$u = 2x, \quad \mathrm{d}u = 2\ \mathrm{d}x$$

$$\mathrm{d}v = sen\ x\ \mathrm{d}x, \quad v = -cos\ x$$

Se sigue:

$$= x^2 sen\ x - \left[-2xcos\ x + 2\int cos\ x\ \mathrm{d}x\right] = x^2 sen\ x + 2xcos\ x - 2sen\ x =$$

$$2xcos\ x + (x^2 - 2)sen\ x \tag{3.2}$$

Por último se obtiene el valor buscado a través de la Regla de Barrow en el intervalo $[1, 3]$. Para ello se empleará el lenguaje de programación interpretado Python, evaluando la expresión 3.2, el resultado se muestra en la tabla 3.1, que se encuentra en la página 10. Se puede ver el código Python en el apéndice A.4

3.1.3. Comparación gráfica.

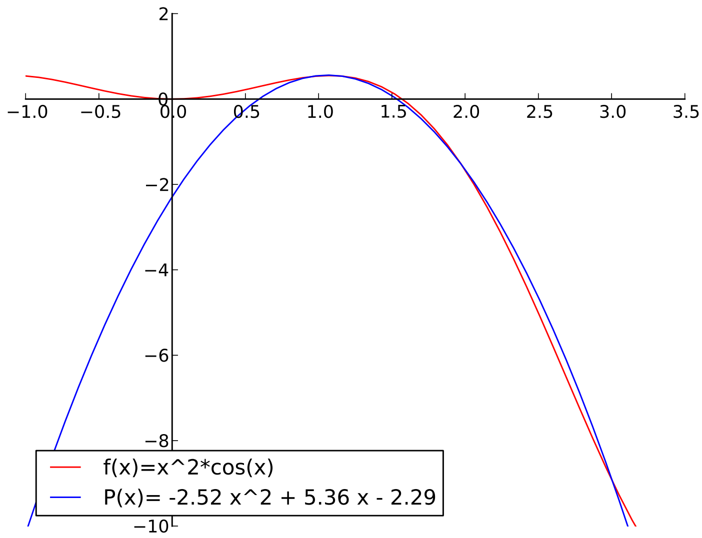

Figura 3.2: Funcion vs. Polinomio interpolador

En esta sección se obtendrá la expresión del polinomio interpolador de la regla de Simpson. Para ello, se busca la expresión del polinomio de grado dos que pasa por los puntos $(1, f(1)), (2, f(2))$ y $(3, f(3))$ de la función 3.1.

De esta forma, tenemos:

$$a1^2 + b1 + c = f(1)$$
$$a2^2 + b2 + c = f(2)$$
$$a3^2 + b3 + c = f(3)$$

Obteniendo como resultado:

$$P_2(x) = -2{,}520227735x^2 + 5{,}355793554x - 2{,}295263513 \quad (3.3)$$

A continuación se implementa en Python un programa que representa la gráfica $P_2(x)$, frente a $f(x)$, como se puede ver en el apéndice A.2, obteniendo como resultado la figura 3.2.

3.1.4. Aproximación por la regla de Simpson.

Para el cálculo de una aproximación de la integral:

$$\int_1^3 x^2 \; cos \; x \; \mathrm{d}x$$

se emplea el lenguaje Python, implementando una función que recibe por parámetros la expresión a estudiar y los extremos del intervalo donde se va a calcular, devolviendo la aproximación buscada, el resultado se muestra en la tabla 3.1, que se encuentra en la página 10. Se puede ver el código Python en el apéndice A.4

3.1.5. Aproximación por la regla de Simpson compuesta.

Para un mejor análisis de los resultados, se repite el estudio aplicando la Regla de Simpson compuesta, implementando una función en Python que recibe por parámetros la expresión a estudiar, los extremos del intervalo y el número par de subintervalos en los que se ha dividido, devolviendo la aproximación buscada, el resultado se muestra en la tabla 3.2, que se encuentra en la página 10. Se puede ver el código Python en el apéndice A.5)

3.2. Descripción del material

Las características del ordenador empleado para la realización de este trabajo son las siguientes:

- CPU type: Intel(R) Atom(TM) CPU N270 @ 1.60GHz
- CPU speed: 800.000Hz
- Cache size: 512 KB

En cuanto al sistema operativo, se ha utlizado “Linux-3.2.0-39-generic-i686-with-Ubuntu-12.04-precise”. Los ficheros en lenguaje Python fueron realizados con el editor de textos avanzado para KDE “Kate”.

3.3. Resultados obtenidos

Valor exacto	Aproximación	Error absoluto	Error relativo
-5.19124855011	-5.0093265161	0.181922034015	0.0350439845558

Cuadro 3.1: Aproximación y error de la regla de Simpson

Número de subintervalos	Aproximación	Error absoluto	Error relativo
2	-5.1817934049	0.0094551452	0.0018213625
6	-5.1911377092	0.0001108409	0.0000213515
10	-5.1912342437	0.0000143064	0.0000027559
50	-5.1912485273	0.0000000228	0.0000000044
100	-5.1912485487	0.0000000014	0.0000000003

Cuadro 3.2: Aproximación y error de la regla de Simpson compuesta

3.4. Análisis de los resultados

Como se puede apreciar en la tabla 3.1, el error cometido al emplear la regla de Simpson, garantiza, para la función estudiada, una precisión de por lo menos una cifra decimal exacta. Por otro lado, se puede observar, que empleando la regla de Simpson compuesta el error es inversamente proporcional al número de intervalos, siendo posible reducirlo tanto como se desee.

Capítulo 4

Conclusiones

- La integración numérica reduce el cálculo del valor de la integral definida en el caso de funciones muy complejas, a simples operaciones aritméticas.
- La Regla de Simpson simplifica el cálculo de una integral definida compleja, aproximando su valor al de la integral de un polinomio de segundo grado.
- La Regla de Simpson es más precisa que otros métodos como la Regla del Trapecio a la hora de aproximar la integral.
- La Regla de Simpson proporciona aproximaciones exactas para polinomios de grado menor o igual a tres.
- La Regla de Simpson Compuesta proporciona aproximaciones más precisas que la Regla de Simpson.
- En la Regla de Simpson Compuesta, el error es inversamente proporcional al número de subintervalos.
- En la Regla de Simpson Compuesta, el error es directamente proporcional a la amplitud del intervalo.

Apéndice A

Ficheros Python

A.1. Grafica de la función

```
#####################################################################################
# Fichero: grafica.py
#####################################################################################
#
# Contenido: Representacion grafica de la funcion f(x)=x**2*cos(x)
#
# Autor/es: Adrian R. Mendioroz Morales
#           Roberto C. Palenzuela Criado
#
#####################################################################################
# Fecha de creacion: 06 de mayo de 2013
#####################################################################################
from math import cos
import matplotlib.pyplot as plt
from matplotlib.ticker import MultipleLocator, FormatStrFormatter
import numpy as np
from matplotlib.pylab import *

def f(t):
  return t**2*cos(t)

t = linspace(-1, 3.5, 51)  # 51 puntos entre 0 y 3
y = zeros(len(t))       # reserva memoria para y con elementos flotantes
for i in xrange(len(t)):
  y[i] = f(t[i])

fig = plt.figure(1)
ax = fig.add_subplot(111)
# set up axis
ax.spines['left'].set_position('zero')
ax.spines['right'].set_color('none')
ax.spines['bottom'].set_position('zero')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.yaxis.set_ticks_position('left')
```

```
plot(t,y)
xlabel('x')
ylabel('y')
legend(['f(x)=x^2*cos(x)'],'best')
axis([-1,3.5, -10, 2])  #[tmin, tmax, ymin, ymax]
savefig('grafica.eps')
show()
```

A.2. Gráfica comparativa

```
####################################################################################
# Fichero: grafica2.py
####################################################################################
#
# Contenido: Representacion grafica de la funcion f(x)=x**2*cos(x) y el polinomio
#            de grado 2 que resulta de alicar el metodo de la regla de Simpson.
#
# Autor/es: Adrian R. Mendioroz Morales
#           Roberto C. Palenzuela Criado
#
####################################################################################
# Fecha de creacion: 06 de mayo de 2013
####################################################################################
from math import cos
import matplotlib.pyplot as plt
from matplotlib.ticker import MultipleLocator, FormatStrFormatter
import numpy as np
from matplotlib.pylab import *

def f1(t):
  return t**2*cos(t)

def f2(t):
  return -2.52*t**2+5.36*t-2.29

t=linspace(-1,3.5,51)
y1 = f1(t)
y2 = f2(t)

fig = plt.figure(1)
ax = fig.add_subplot(111)
# set up axis
ax.spines['left'].set_position('zero')
ax.spines['right'].set_color('none')
ax.spines['bottom'].set_position('zero')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.yaxis.set_ticks_position('left')

plot(t,y1,'r')
hold('on')
plot(t,y2,'b')
```

```
legend(['f(x)=x^2*cos(x)', 'P(x)= -2.52 x^2 + 5.36 x - 2.29'],'best')
axis([-1,3.5, -10, 2])  #[tmin, tmax, ymin, ymax]
savefig('grafica2.eps')
show()
```

A.3. Modulo Simpson

```
#################################################################################
# Fichero: simpson.py
##################################################################################
# Modulo: Simpson
#
# Contenido: Funciones de aplicacion de la regla de Simpson de integracion numerica.
#
# Autor/es: Adrian R. Mendioroz Morales
#          Roberto C. Palenzuela Criado
##################################################################################
# Funciones:
#   regla_simpson: Calcula una aproximacion a la integral de una funcion en un
#                  intervalo [a,b], aplicando la regla de Simpson.
#      Entrada:
#         f: funcion que se va a aproximar
#         a: limite inferior de integracion
#         b: limite superior de integracion
#      Salida:
#         Aproximacion de la integral de f en [a,b]
#
#   regla_simpson_compuesta: Calcula una aproximacion a la integral de una funcion en un
#                            intervalo [a,b], empleando n subintervalos y aplicando la
#                            regla de Simpson compuesta.
#      Entrada:
#         f: funcion que se va a aproximar
#         a: limite inferior de integracion
#         b: limite superior de integracion
#         n: numero (par) de subintervalos
#      Salida:
#         Aproximacion de la integral de f en [a,b]
##################################################################################
# Fecha de creacion: 06 de mayo de 2013
##################################################################################

from math import *

def regla_simpson(f,a,b):
  return (b-a)/6.0*(f(a)+4*f((a+b)/2.0)+f(b))

def regla_simpson_compuesta(f,a,b,n):
  h=(b-a)/float(n)
  aprox=0
  for l in range(0,n):
    aprox+=regla_simpson(lambda x:(x**2*cos(x)),a+l*h,a+(l+1)*h)
  return aprox
```

A.4. Aproximacion y error de la regla de Simpson

```
#! /usr/bin/python
#####################################################################################
# Fichero: uso_simpson.py
#####################################################################################
#
# Contenido: Comprobacion del error cometido al aplicar la regla de Simpson
#            al calculo de la integral de  la funcion f(x)=x**2*cos(x),
#            en el intervalo [a,b].
#
# Autor/es: Adrian R. Mendioroz Morales
#           Roberto C. Palenzuela Criado
#
#####################################################################################
# Fecha de creacion: 06 de mayo de 2013
#####################################################################################
from simpson import regla_simpson
from math import *

F=(2*3*cos(3)+(3**2-2)*sin(3))-(2*1*cos(1)+(1**2-2)*sin(1))
print '\nValor real= ',F

aprox=regla_simpson(lambda x:(x**2*cos(x)),1,3)
print '\nAproximacion por la regla de Simpson: ',aprox
print '\t\t      Error absoluto: ', abs(F-aprox)
print '\t\t      Error relativo: ', abs(F-aprox)/abs(F)
print "\n"
```

A.5. Aproximacion y error de la regla de Simpson

```
#! /usr/bin/python
#####################################################################################
# Fichero: uso_simpson_compuesta.py
#####################################################################################
#
# Contenido: Comprobacion de los errores cometidos al aplicar la regla de Simpson
#            compuesta al calculo de la integral de la funcion f(x)=x**2*cos(x),
#            en el intervalo [a,b].
#
# Autor/es: Adrian R. Mendioroz Morales
#           Roberto C. Palenzuela Criado
#
#####################################################################################
# Fecha de creacion: 06 de mayo de 2013
#####################################################################################
from simpson import regla_simpson_compuesta
from math import *

F=(2*3*cos(3)+(3**2-2)*sin(3))-(2*1*cos(1)+(1**2-2)*sin(1))
print '\nValor real= ',F

print '\nAproximacion por la regla de Simpson compuesta: '
```

```
print '%25s %15s %15s %15s' % ('Numero de subintervalos','Aproximacion','Error absoluto',
'Error relativo')
n=[2,6,10,50,100]
aprox=0
i=0
for l in n:
  aprox=regla_simpson_compuesta(lambda x:(x**2*cos(x)),1,3,l)
  print '%16d %24.10f %15.10f %15.10f' %(n[i],aprox,abs(F-aprox),abs(F-aprox)/abs(F))
  i=i+1
print "\n"
```

Bibliografía

[1] David Kincaid and Ward Cheney. *Análisis Numérico. Las matemáticas del cálculo científico.* Addison-Wesley Iberoamericana, 1994.

[2] Leslie Lamport. *LaTeX: A Document Preparation System.* Addison–Wesley Pub. Co., Reading, MA, 1986.

[3] Jacqueline Lelong-Ferran and J.M. Arnaudiés. *Curso de Matemáticas. Tomo 2. Análisis.* Editorial Reverte, S.A., 1982.

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

[5] Lino J. Alvarez Vazquez and Aurea M. Martinez Varela. *Lecciones de Métodos Numéricos.* SPTV, 2005.

[6] Eric W. Weisstein. http://mathworld.wolfram.com/SimpsonsRule.html.