Actividad 9

Andrés Ignacio Rodríguez Mendoza

Introducción

En la actividad 6 se resolvió numericamente la integral del periodo para cualquier ángulo inicial del péndulo:

$$T = 4\sqrt{\frac{l}{2g}}\int_0^{\theta_0} \frac{1}{\sqrt{\cos\theta - \cos\theta_0}} d\theta.$$

Esta integral se puede reescribir en términos de una integral elíptica como

$$T = a\sqrt{\frac{l}{g}} F\left(\theta_0/2, \csc\frac{\theta_0}{2}\right)\csc\frac{\theta_0}{2}$$

donde F es la integral elíptica incompleta de primer tipo, defnida por

$$F(\phi, k) = \int_0^{\phi} \frac{1}{\sqrt{1 - k^2 \sin^2 u}} du.$$

Más concisamente por la sustitución $\sin u = \sin(\theta/2)/(\sin(\theta_0/2)$ expresando θ en términos de u,

$$T = 4\sqrt{\frac{l}{g}} K\left(\sin^2\frac{\theta_0}{2}\right)$$

donde K es una integral completa de primer tipo definida por

$$K(k) = F(\pi/2, k) = \int_0^{\pi/2} \frac{1}{\sqrt{1 - k^2 \sin^2 u}} du.$$

La solución a la integral elíptica son los polinomios de Legendre y el periodo del péndulo es

$$T = 2\pi\sqrt{\frac{l}{g}} \sum_{n=0}^{\infty} \left[\left(\frac{(2n)!}{(2^n \cdot n!)^2}\right)^2 \cdot \sin^{2n}\left(\frac{\theta_0}{2}\right)\right]$$

En esta práctica se calculan los coeficientes de esta serie y se compara en las gráficas el valor de la integral con la aproximación en distintos órdenes del polinomio. Se muestra que también puede encontrarse la solución mediante la serie de Maclaurin

$$\sin\frac{\theta_0}{2} = \frac{1}{2}\theta_0 - \frac{1}{48}\theta^3 + \frac{1}{3840}\theta_0^5 - \frac{1}{645120}\theta_0^7 + \cdots$$

sustituyendo esta serie en los polinomios de Legendre.

Código

Código 1: Código para calcular errores

```
import numpy as np
from scipy import integrate
import matplotlib.pyplot as plt
```

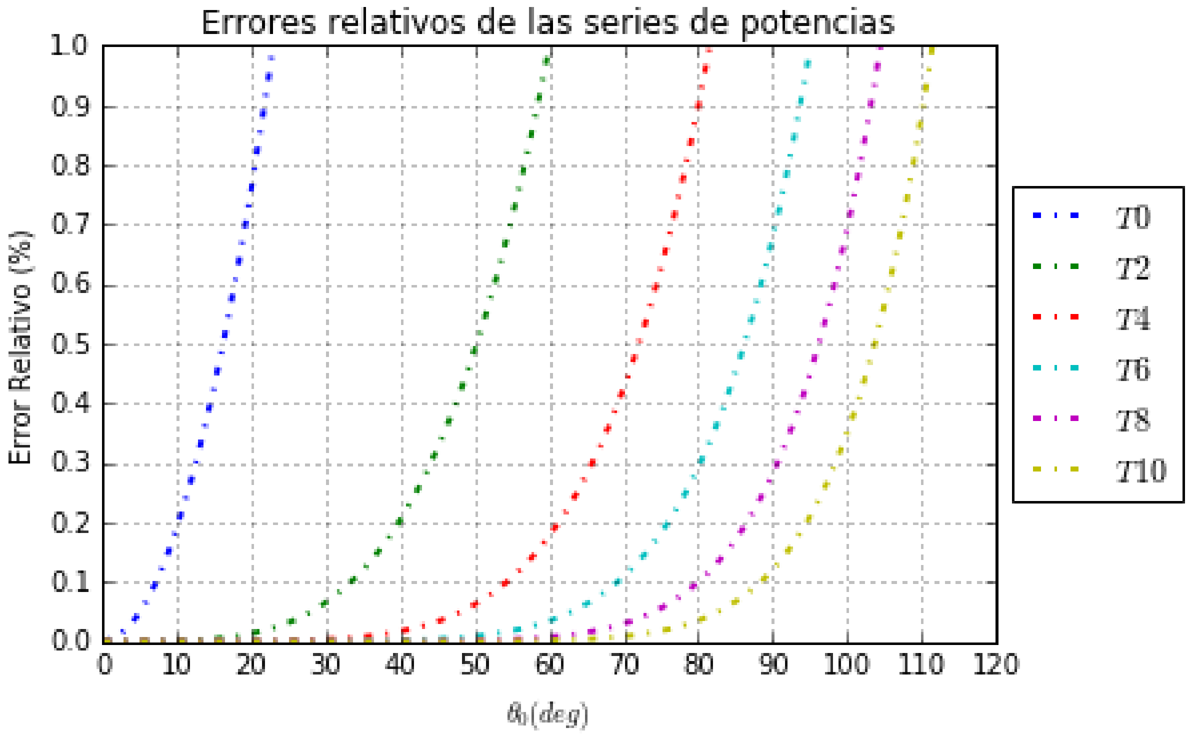
```
import math

t=100
n=6
# Arreglos
x=[]
TT0_0=[]
TT0=[]
x_0=np.linspace(0.001,np.pi + 0.001, t)

#   el integrando
I = lambda x,a: 1/np.sqrt(np.cos(x) - np.cos(a))

#   Calcular los valores reales
for i in range(t):
#   la integral
    theta_0=x_0[i]
    T_0 , err= integrate.quad(I, 0, theta_0, args=(theta_0,))


#   el periodo
    TT0_0.append((4/np.sqrt(2)) * T_0)


#   Ciclo para cada gráfica:
for v in range(n):

#   lista de los valores de error
    err=[]

    for i in range(t):


        theta_0 = x_0[i]
        T0=1

#   la sumatoria
        for u in range(v):

            T0 += math.pow( math.factorial(2*(u+1)) / (math.pow( math.pow(2,(u+1)) * ma

#   los valores en la lista de errores
        err.append(100*(np.absolute(2 * np.pi * T0 - TT0_0[i])/TT0_0[i]))

#   Gráfica, cada aproximación
    plt.plot(x_0 * 180 / np.pi, err, '-.', linewidth=2, label='$T_%i_$' %(2*v))
```

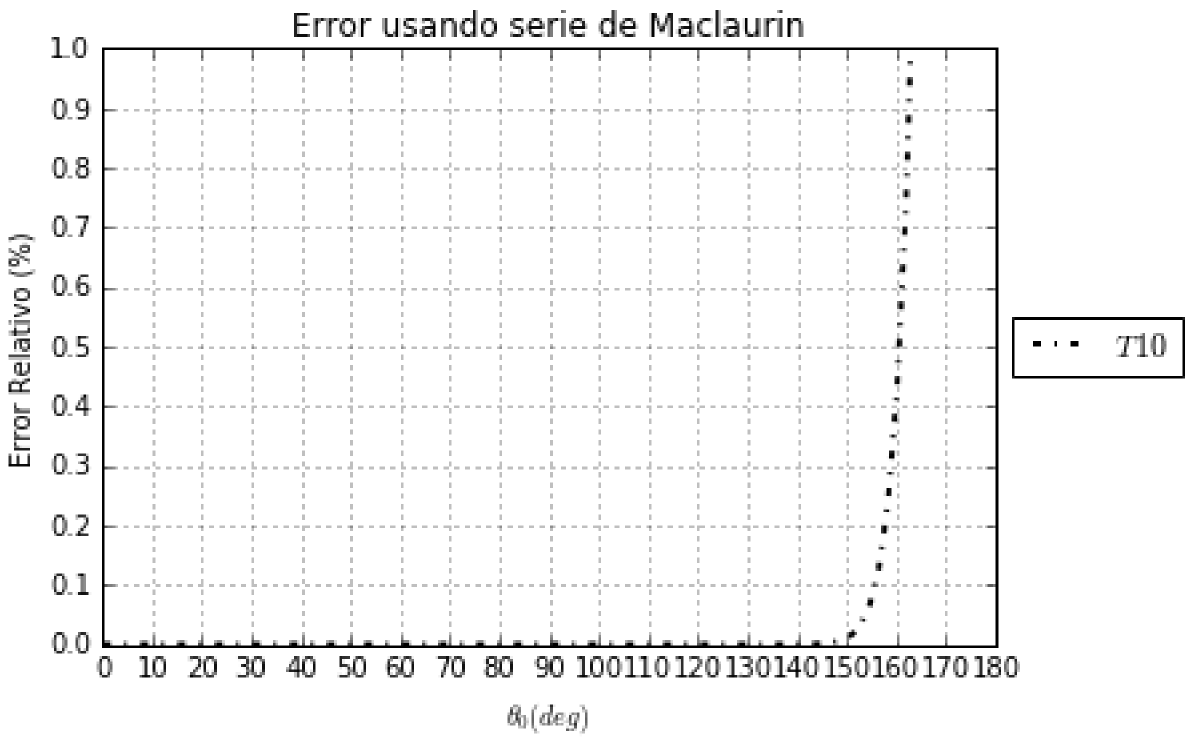
```
#   Gráfica desviación
plt.title('Errores␣relativos␣de␣las␣series␣de␣potencias')
plt.xlabel(r'$␣\theta␣␣0␣(deg)$')
plt.ylabel("Error␣Relativo␣(%)")
plt.xlim(0,120)
plt.ylim(0,1)
plt.xticks(np.arange(0,130,10))
plt.yticks(np.arange(0,1.1,0.1))
plt.legend(loc='center␣left', bbox_to_anchor=(1, 0.5))
plt.grid()
plt.show()
```

Código 2: Segmento de código para la serie de Maclaurin

```
Err=[]
for i in range(t):
    theta_0 = x_0[i]
    T0=1
    for u in range(80):
        sen=0
# la sumatoria para maclaurin en el seno
        for k in range(50):
            sen += math.pow(-1,k)/math.factorial(2*k+1) * math.pow(theta_0/2, 2*k+1)

        T0 += math.pow( math.factorial(2*(u+1)) / (math.pow( math.pow(2,(u+1)) * math.f

    Err.append(100*(np.absolute(2 * np.pi * T0 - TT0_0[i])/TT0_0[i]))

plt.plot(x_0 * 180 / np.pi, Err, '-.',color='k', linewidth=2, label='$T␣%i␣$' % (2*v))
plt.title('Error␣usando␣serie␣de␣Maclaurin')
plt.xlabel(r'$␣\theta␣␣0␣(deg)$')
plt.ylabel("Error␣Relativo␣(%)")
plt.xlim(0,180)
plt.ylim(0,1)
plt.xticks(np.arange(0,190,10))
plt.yticks(np.arange(0,1.1,0.1))
plt.legend(loc='center␣left', bbox_to_anchor=(1, 0.5))
plt.grid()
plt.show()
```

Gáficas


Errores relativos de las series de potencias
Error Relativo (%)
$\theta_0(deg)$
T0
T2
T4
T6
T8
T10


Error usando serie de Maclaurin
Error Relativo (%)
$\theta_0(deg)$
T10