

Margarita Pérez Ragazzo

Curso: Medotos Matematicos

Tarea: Analizar dos series armonicass, verificar si convergen o divergen.



Descripción del Código

Este código en Python realiza un análisis numérico de dos series matemáticas:

1. **Serie armónica:** $\sum(1/n)$ desde $n=1$ hasta 10^6
2. **Serie de inversos de cuadrados:** $\sum(1/n^2)$ desde $n=1$ hasta 10^6

El programa calcula las sumas parciales de estas series y determina si convergen o divergen, comparando los resultados con los valores teóricos conocidos. Además, genera gráficas que muestran visualmente el comportamiento de ambas series.

Características principales

- Cálculo numérico de sumas parciales hasta $n = 1,000,000$
- Determinación de convergencia/divergencia basada en el comportamiento numérico
- Visualización gráfica del crecimiento de las series
- Comparación con valores teóricos ($\pi^2/6$ para la serie de inversos de cuadrados)

Resultados esperados

El programa generará:

1. Valores numéricos de las sumas parciales
2. Determinación de convergencia/divergencia
3. Dos gráficas mostrando el comportamiento de las series:
 - Serie armónica (crecimiento logarítmico, divergente)
 - Serie de inversos de cuadrados (convergencia a $\pi^2/6$)

The screenshot shows a Windows 10 desktop with a VS Code editor window open. The editor is displaying a Python script named `sumatoriapy.py` in the `sumatoriapy 2` editor. The script is a numerical analysis program that calculates the sum of an arithmetic series and its convergence. The script is as follows:

```

1  """
2  Código para el análisis numérico de convergencia de series
3  Elaborado para el curso de Métodos Computacionales en Física
4  """
5
6  import numpy as np
7  import matplotlib.pyplot as plt
8
9  # Configuración de parámetros
10 n_max = 1000000 # 10^6 términos
11 muestras_grafico = 1000 # Puntos para visualización
12
13 print("Iniciando cálculo de series...")
14 print(f"Número de términos: {n_max}")
15 print(f"n = {n_max} * 50")
16
17 # Serie armónica: 1/(1/n)
18 print("Calculando serie armónica...")
19 suma_armonica = 0.0
20 valores_n = np.logspace(1, np.log10(n_max), muestras_grafico).astype(int)
21 valores_armonica = []
22
23 for n in range(1, n_max + 1):
24     suma_armonica += 1.0 / n
25

```

The terminal window at the bottom shows the output of the script, indicating that the series is divergent:

```

ANÁLISIS:
La serie armónica muestra crecimiento continuo → DIVERGENTE
La serie de inversos cuadrados tiende a π²/6 → CONVERGENTE

```

The taskbar at the bottom shows the Windows Start button, a search bar, and several application icons including VS Code, File Explorer, and various utility tools. The system tray shows the date and time as 01:51 p.m. on 15/09/2025.

The figure consists of two side-by-side plots. The left plot is titled 'Serie Armónica: $\Sigma(1/n)$ '. The y-axis is labeled 'Suma parcial S_n ' and ranges from 4 to 14. The x-axis is labeled 'n (escala log)' and ranges from 10^1 to 10^6 . A blue line shows the partial sums increasing linearly on this log scale. The right plot is titled 'Serie: $\Sigma(1/n^2)$ '. The y-axis is labeled 'Suma parcial S_n ' and ranges from 1.56 to 1.64. The x-axis is labeled 'n (escala log)' and ranges from 10^1 to 10^6 . A red curve shows the partial sums increasing and then leveling off, approaching a horizontal dashed line at $\pi^2/6 \approx 1.6449$.

Prints:

Para la serie armónica

```
sumatoria.py 2 X
sumatoria.py > ...
1
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
n=803358: Sn ≈ 14.173762
n=812661: Sn ≈ 14.185286
n=822081: Sn ≈ 14.196810
n=831610: Sn ≈ 14.208335
n=841249: Sn ≈ 14.219859
n=851000: Sn ≈ 14.231384
n=860864: Sn ≈ 14.242908
n=870843: Sn ≈ 14.254433
n=880937: Sn ≈ 14.265958
n=891148: Sn ≈ 14.277482
n=901477: Sn ≈ 14.289006
n=911926: Sn ≈ 14.300530
n=922497: Sn ≈ 14.312056
n=933189: Sn ≈ 14.323579
n=944006: Sn ≈ 14.335104
n=954948: Sn ≈ 14.346628
n=966017: Sn ≈ 14.358153
n=977214: Sn ≈ 14.369677
n=988541: Sn ≈ 14.381202
n=1000000: Sn ≈ 14.392727

Calculando serie de inversos cuadrados...
n=10: Sn ≈ 1.549768
n=11: Sn ≈ 1.558032
n=12: Sn ≈ 1.564977
n=13: Sn ≈ 1.570894
n=14: Sn ≈ 1.575996
n=15: Sn ≈ 1.580440
n=16: Sn ≈ 1.584347
n=17: Sn ≈ 1.587807
n=18: Sn ≈ 1.590893
n=19: Sn ≈ 1.593663
n=20: Sn ≈ 1.596163
```

Serie del cuadrado inverso:

```
sumatoria.py 2 X
sumatoria.py > ...
1
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
n=715904: Sn ≈ 1.644933
n=724202: Sn ≈ 1.644933
n=732596: Sn ≈ 1.644933
n=741088: Sn ≈ 1.644933
n=749678: Sn ≈ 1.644933
n=758367: Sn ≈ 1.644933
n=767158: Sn ≈ 1.644933
n=776050: Sn ≈ 1.644933
n=785045: Sn ≈ 1.644933
n=794145: Sn ≈ 1.644933
n=803358: Sn ≈ 1.644933
n=812661: Sn ≈ 1.644933
n=822081: Sn ≈ 1.644933
n=831610: Sn ≈ 1.644933
n=841249: Sn ≈ 1.644933
n=851000: Sn ≈ 1.644933
n=860864: Sn ≈ 1.644933
n=870843: Sn ≈ 1.644933
n=880937: Sn ≈ 1.644933
n=891148: Sn ≈ 1.644933
n=901477: Sn ≈ 1.644933
n=911926: Sn ≈ 1.644933
n=922497: Sn ≈ 1.644933
n=933189: Sn ≈ 1.644933
n=944006: Sn ≈ 1.644933
n=954948: Sn ≈ 1.644933
n=966017: Sn ≈ 1.644933
n=977214: Sn ≈ 1.644933
n=988541: Sn ≈ 1.644933
n=1000000: Sn ≈ 1.644933

=====
RESULTADOS FINALES
```

Análisis:

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

n=966017:  $S_n \approx 1.644933$ 
n=977214:  $S_n \approx 1.644933$ 
n=988541:  $S_n \approx 1.644933$ 
n=1000000:  $S_n \approx 1.644933$ 

=====
RESULTADOS FINALES
=====

Serie armónica ( $\sum 1/n$ ):  $S \approx 14.392727$ 
Serie de inversos cuadrados ( $\sum 1/n^2$ ):  $S \approx 1.644933$ 
Valor teórico ( $\pi^2/6$ ): 1.644934
Valor teórico ( $\pi^2/6$ ): 1.644934

ANÁLISIS:
La serie armónica muestra crecimiento continuo → DIVERGENTE
La serie de inversos cuadrados tiende a  $\pi^2/6$  → CONVERGENTE
```

Código en c++:

Para este igual utilicé el visual estudio code. Resultado de la terminal y el csv:

```
File Edit Selection View Go Run ... metodos_mat
series.cpp datos_series.csv series.exe analisis_series.png
EXPLORER
  OPEN EDITORS
    series.cpp
    datos_series.csv output
    series.exe output
    analisis_series.png
  METODOS_MAT
    output
      datos_series.csv
      series.exe
      analisis_series.png
      series.cpp
      sumatoria.py
  OUTPUT
  TIMELINE
  SEARCH
  OUTPUT
  DEBUG CONSOLE
  TERMINAL
  PORTS
  C/C++ Compile Run
  Valor teorico  $\pi^2/6$ : 1.64493
  PS C:\metodos_mat\output> cd "C:\metodos_mat\output"
  PS C:\metodos_mat\output> & .\series.exe
  Calculando series hasta n = 1000000...
  Procesado: 1000000/1000000 terminos
  ¡Cálculo completado!
  Resultados guardados en 'datos_series.csv'
  Valor final serie armónica: 14.3927
  Valor final serie inversos cuadrados: 1.64493
  Valor teórico  $\pi^2/6$ : 1.64493
  PS C:\metodos_mat\output>
```