



**I  
N  
A  
O  
E**

**Instituto Nacional de Astrofísica, Óptica,  
y Electrónica.**

**Reporte Técnico**

**No. 687**

COORDINACIÓN DE ELECTRÓNICA

**Entropía Kolmogorov-Sinaí:  
Análisis y un caso de aplicación**

Dr. Pedro Pancóatl Bortolotti  
Dr. Rogerio Enríquez Caldera\*

\*INAOE/CRECTEALC

20 marzo del 2024

©INAOE 2024

Derechos reservados

El autor responsable otorga al INAOE el permiso de reproducir y distribuir copias de este reporte técnico en su totalidad o en partes mencionando la fuente



# Entropía Kolmogorov-Sinaí: Análisis y un caso de aplicación

## Sumario

Entropía de Kolmogorov.....	2
Entropía de Kolmogorov-Sinaí.....	4
Consideraciones y estimación de entropía de K-S.....	5
Entropía de Kolmogorov del mapa Henon.....	6
Referencias.....	7
Anexos.....	7

## Abstract

*The document presents a comprehensive analysis of the Kolmogorov-Sinaí metric, briefly explaining the background, interpretation, and needs to identify behaviors in either deterministic or stochastic dynamic systems. In the end, a particular application case is exposed to identify chaos in the discrete Henón map.*

*Palabras clave: Kolmogorov-Sinaí, mapa Henón, caos, sistemas dinámicos, simulación*

*Keywords: Kolmogorov-Sinaí, Henón map, chaos, dynamic systems, simulation*

## Entropía de Kolmogorov

Dentro del estudio de sistemas dinámicos, la métrica de Kolmogorov-Sinaí que se deriva de la entropía de Shannon en la teoría de la información y es adaptada para demostrar si dos sistemas dinámicos no están conjugados para permitir una definición cualitativa de caos. Por tal motivo, la entropía de K es una medida importante mediante el cual un movimiento caótico en el espacio fase puede ser caracterizado. También, está asociada al incremento del desorden como lo establece la entropía termodinámica, que además expresa el incremento del grado de incertidumbre o ignorancia acerca del estado del sistema.

El físico austriaco Ludwig Boltzmann, contribuyó con el desarrollo clave en la denominada “entropía”, cuando aplica para la mecánica estadística la medida estadística de la entropía

$$H = -K \sum_{i=1}^N P_i \log P_i, \quad (1)$$

donde K es la constante de Boltzmann, y  $P_i$  la probabilidad de un evento que se encuentra en uno de los  $N$  estados del espacio fase de un sistema (Williams, 1998).

Por su parte Shannon, definió la entropía como una medida de información, que a diferencia de la ecuación (1), pudo aplicar a muchos tipos de probabilidad. Actualmente aún se discute si la entropía de Shannon mide la misma cosa que la entropía de Boltzmann, aunque la realidad es que la información y la entropía están siendo tratadas como una misma cosa, situación que también ocurre en **caos**.

Así, la forma modificada de la ecuación (1) que involucra entropía como información es

$$H_w = \sum_{i=1}^N P_i \log(1/P_i), \quad (2)$$

ponderada como un valor promedio, denominada por muchos autores como **entropía de información**.

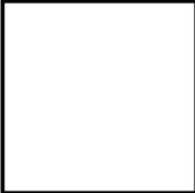
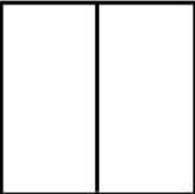
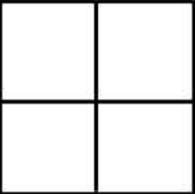
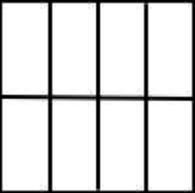
La ecuación Tabla 1 describe algunas interpretaciones y estados opuestos para la entropía tal como se muestran en la Tabla 1 a continuación:

Tabla 1. Estados opuestos de entropía e interpretación

<b>Alta Entropía</b>	<b>Baja Entropía</b>
Grande porción de energía disponible para realizar un trabajo	Pequeña porción de energía disponible para realizar un trabajo
Desorden, desorganización, mezcla completa	Orden, alto grado de organización, ordenamiento meticoloso
Eventos igualmente probables, baja probabilidad de un evento seleccionado	Resultados preordenados, alta probabilidad de un evento seleccionado
Distribución uniforme	Distribución altamente desigual
Gran incertidumbre	Altamente confiable, certero
Aleatoriedad o imprevisibilidad	No aleatoriedad, pronóstico preciso
Libertad de elección, muchos posibles resultados	Elección restringida estrechamente, pocos resultados posibles
Larga diversidad	Poca diversidad
Gran sorpresa	Poca o nula sorpresa
Mucha información	Poca información
Gran cantidad de información usada para especificar el estado del sistema	Pequeña cantidad de información usada para especificar el estado del sistema
Alta precisión de datos	Baja precisión de datos

El punto resaltado en letras negritas de la Tabla 1, interpreta la entropía como la cantidad de información necesaria para especificar el estado del sistema con una precisión particular. De este modo, si se considera la subdivisión del espacio fase a manera de contenedores de tamaño  $\epsilon$  como se muestra en la Tabla 2, entonces cada división o compartimiento provee la misma probabilidad para el estado del sistema.

Tabla 2. Probabilidades e información para resultados igualmente probables

Rejilla	Número $N_s$ de compartimentos o estados	Probabilidad $P=1/N_s$	Entropía $H = \log_2(1/P) = \log_2 N_s$ (bits)
	<b>1</b>	<b>1</b>	<b>0</b>
	<b>2</b>	<b>0.5</b>	<b>1</b>
	<b>4</b>	<b>0.25</b>	<b>2</b>
	<b>8</b>	<b>0.125</b>	<b>3</b>

### Entropía de Kolmogorov-Sinai

La entropía de Shannon (entropía de información) no puede por sí misma identificar caos (tómese como ejemplo el caso binario de la Tabla 2), ya que no depende de parámetros de control, número de compartimentos, y tamaño entre otros factores. Por tanto, las ecuaciones (1) y (2) forman la base de otro tipo de entropía, que teóricamente puede identificar caos, esta es llamada entropía *Kolmogorov-Sinai* (*K-S*), también conocida como *entropía teórica* o *entropía métrica*, la cual tiene tres características importantes:

1. Dado que el sistema seguirá varias rutas sobre el tiempo, K-S requiere una secuencia temporal de probabilidades.

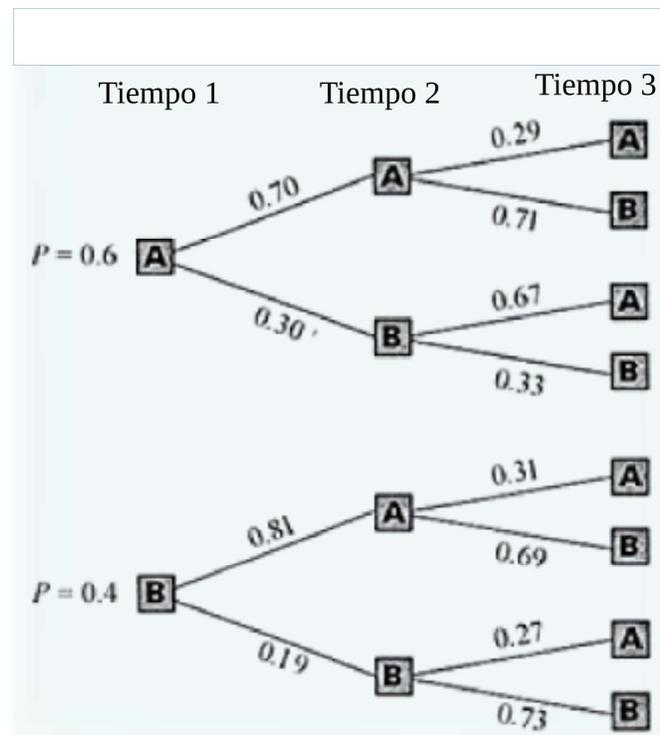
2. K-S representa la ‘tasa’ de entropía promedio en el tiempo como  $H = \sum_{i=1}^{N_s} P_s \log(1/P_s) / (\text{tiempo})$ .
3. La entropía K-S es un valor de límite cuando el tiempo tiende a infinito, y el tamaño  $\epsilon$  del contenedor tiende a cero.

Basado en estas tres características, la **entropía K-S** ( $H_{KS}$ ) es la entropía promedio por unidad de tiempo en las condiciones límites de tiempo y tamaños de contenedor - también denominada caja - y está dada por

$$H_{KS} = \lim_{\epsilon \rightarrow 0} \lim_{t \rightarrow \infty} \left[ \sum_{i=1}^N P_s \log(1/P_s) \right] / (\text{tiempo}), \quad (3)$$

En particular, la característica 1 arriba mencionada trata con la información o incertidumbre asociada con una secuencia de tiempo de mediciones u observaciones, para entenderlo considerar el diagrama de la Figura 1, para dos posibles estados **A** y **B**.

Figura 1. Patrón de ramas y probabilidades hipotéticas para dos contenedores A y B



En el tiempo 1, hay sólo dos posibles rutas hacia los contenedores A y B con probabilidades ordinarias de 0.6 y 0.4 respectivamente. De este modo, y considerando el logaritmo de base dos por simplicidad, para el compartimiento A,  $P \log_2(1/P) = 0.6 \log_2(1/0.6) = 0.442$ , para B,  $P \log_2(1/P) = 0.4 \log_2(1/0.4) = 0.529$ , así de acuerdo con la ecuación (2), la entropía es la suma de todas las rutas  $H_{\Delta t} = 0.442 + 0.529 = 0.971$ .

Ahora bien, para el tiempo 2 hay cuatro posibles rutas y probabilidades asociadas AA ( $P_s = 0.6 * 0.7 = 0.420$ ), AB ( $P_s = 0.6 * 0.3 = 0.180$ ), BA ( $P_s = 0.324$ ), BB ( $P_s = 0.076$ ), lo que da la entropía para el tiempo 2 como  $H_{\Delta t} = 0.526 + 0.445 + 0.527 + 0.283 = 1.781$ .

Y así sucesivamente.

La secuencia de probabilidades en análisis de caos usualmente está basada en valores retrasados de una sola variable, siendo posible variar el retraso mismo y el número de dimensiones del espacio fase. Así, algunas interpretaciones de la entropía K-S es :

1. Cantidad promedio de incertidumbre en la predicción de los siguientes  $n$  eventos (Young,1983).
2. Tasa promedio en la cual la precisión de una predicción decae como incrementa el tiempo de predicción, esto es, la tasa en la cual la predictibilidad estará perdida (Farmer, 1982).
3. La tasa promedio en la cual la información acerca del estado del sistema está perdido (Schuster 1988).

### Consideraciones y estimación de entropía de K-S

La ecuación (3) indica la operación para elegir el tamaño del contenedor  $\epsilon$ , de tal modo que al mantener este valor  $\epsilon$  constante, observamos lo que ocurre con la tasa de entropía cuando el tiempo tiende hacia infinito, los primeros pasos del cálculo para tal propósito son:

1. Elegir el ancho de contenedor  $\epsilon$ .
2. Usar datos básicos para estimar la secuencia de probabilidades para todas las posibles rutas, para tiempos sucesivos.
3. Para cada tiempo calcular la entropía como 
$$\sum_{i=1}^N P_s \log(1/P_s)$$

Otras variantes de la ecuación (3) presentada por (Grassberger, 1983), (Zmeskal et al., 2013) permite la obtención de entropía desde series de tiempos sucesivos. Considere por ejemplo, un sistema dinámico de 'd' grados de libertad, y suponer que el espacio fase de  $d$  dimensiones es particionado en contenedores o cajas de tamaño  $\epsilon^d$ , y etiquetado  $1,2,3,\dots, M(\epsilon)$ . Ahora, considerar una serie de tiempo  $\{\bar{x}(i+1), \bar{x}(i+2), \dots, \bar{x}(i+l)\}$  de  $l$  puntos sucesivos, que representan el estado de la trayectoria en intervalos de tiempo  $\Delta T$ , por ejemplo,  $\bar{x}(t_0+i\Delta t)$ . Suponer que la primera muestra se encuentra dentro de la caja  $i(1)$ , el segundo en  $i(2)$ , etc., por lo que la serie de puntos dirige a la secuencia de cajas  $I_l = \{i_1, i_2, \dots, i_l\}$ . Así, el número total de secuencias distintas  $I_l$  que podría ocurrir es  $M(\epsilon)^l$ .

Se ha de considerar que  $P(I_l)$  es la probabilidad de una  $I_l$  dada, donde  $\sum_{I_l} P(I_l) = 1$ .

La entropía de Kolmogorov es entonces:

$$K = - \lim_{\epsilon \rightarrow 0} \lim_{l \rightarrow \infty} \lim_{\Delta t \rightarrow 0} \left( \frac{1}{\Delta T} \right) \sum_{I_l} P(I_l) \ln \left( \frac{P(I_l)}{P(I_{l-1})} \right), \quad (4)$$

que es una ecuación particular presentada por (Termonia, 1984) para la obtención de la entropía K utilizando series de tiempo, y cuya eficiencia puede verificarse a través de la conjetura de la entropía K como la suma de los exponentes positivos de Lyapunov.

La interpretación de la entropía  $K$  es:  $K=0$  para un sistema ordenado,  $K \rightarrow \infty$  en un sistema aleatorio, y  $K \neq 0$  una constante para un sistema caótico determinista.

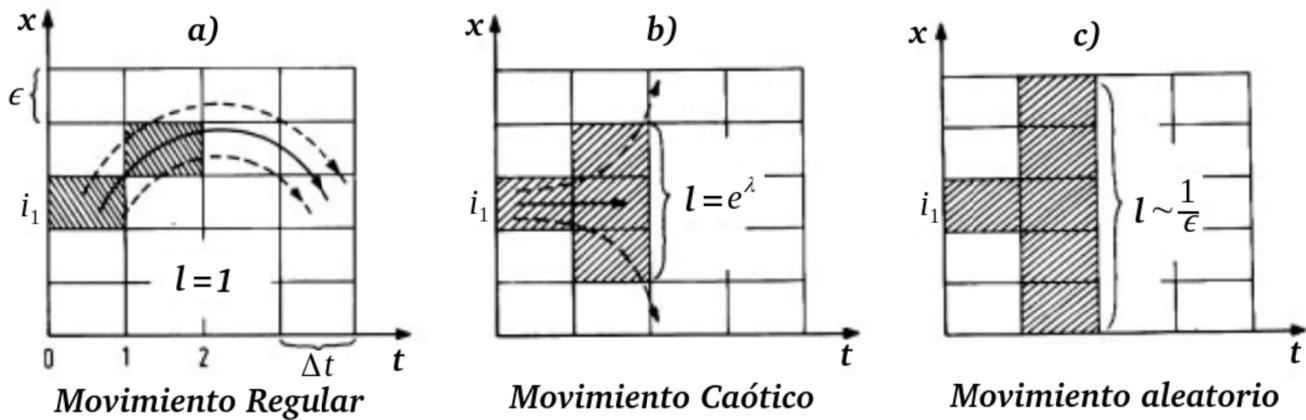


Figura 2.  $K$ -entropías para un movimiento unidimensional.

En la *Figura 2a*, los puntos inicialmente adyacentes permanecen adyacentes, de modo que  $p_{i_1} = \epsilon$ ,  $p_{i_1 i_2} = \epsilon \cdot 1$  por lo que la entropía es  $K=0$ . Por el contrario, en la *Figura 2b*, los puntos inicialmente adyacentes llegan a estar separados exponencialmente, de modo que  $p_{i_1} = \epsilon$ , y  $p_{i_1 i_2} = \epsilon \cdot e^\lambda$ , por lo que  $K=\lambda>0$ . Finalmente, en la *Figura 2c*, los puntos inicialmente adyacentes están distribuidos con igual probabilidad sobre todos los intervalos permitidos, por lo que  $p_{i_1} = \epsilon$ ,  $p_{i_1 i_2} \propto \epsilon^2$ , de modo que  $K \propto -\log \epsilon \rightarrow \infty$  (Schuster & Just, 2005).

### Entropía de Kolmogorov del mapa Henon

Considere el mapa discreto Henon descrito por la ecuación de diferencias

$$\begin{aligned} x_{i+1} &= 1 + y_i - a x_i^2 \\ y_{i+1} &= b x_i, \end{aligned} \tag{5}$$

con parámetros típicos  $a=1.4$ ,  $b=0.3$ , cuya solución produce el atractor Henon de la *Figura 4*.

El espacio fase es segmentado en contenedores  $\epsilon$  con un tamaño pequeño  $\sim 0.05$ , tal que se realiza la medición de la frecuencia de ocurrencias en la que el estado del sistema se encuentra en cada contenedor, de modo que se obtiene un histograma de ocurrencias  $P_i$  del  $i$ -ésimo elemento de la partición, que al normalizarlo para que  $\sum_i P_i = 1$ , se obtiene la *distribución de probabilidad asintótica de grano grueso* (Farmer, 1982).

La mayor distribución de probabilidad puede identificarse en *Figura 3*

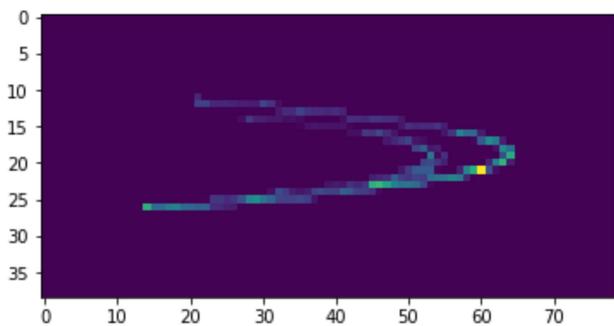


Figura 3. Distribución de probabilidad para mapa Henon

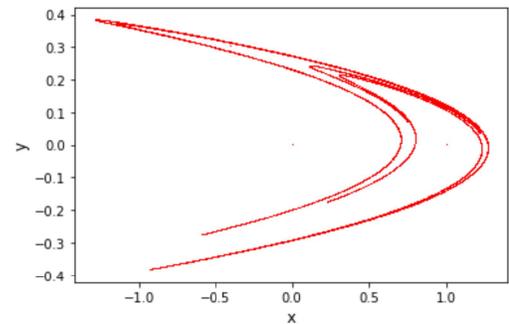


Figura 4. Atractor en espacio fase para mapa Henon

Quedando claramente establecido el movimiento correspondiente y los coeficientes de Lyapunov.

## Referencias

- Farmer, J. D. (1982). Information Dimension and the Probabilistic Structure of Chaos. In *Zeitschrift fur Naturforschung - Section A Journal of Physical Sciences* (Vol. 37, Issue 11, pp. 1304–1326). <https://doi.org/10.1515/zna-1982-1117>
- Grassberger, P. (1983). Estimation of the Kolmogorov entropy from a chaotic signal. *Physical Review*, 28(4), 2591–2593.
- Schuster, H. G., & Just, W. (2005). Deterministic chaos. An Introduction. In *British Homoeopathic Journal* (Fourth, Vol. 82, Issue 2). Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim. [https://doi.org/10.1016/S0007-0785\(05\)81055-5](https://doi.org/10.1016/S0007-0785(05)81055-5)
- Termonia, Y. (1984). Kolmogorov entropy from a time series. *Physical Review A*, 29(3), 1612–1614. <https://doi.org/10.1103/PhysRevA.29.1612>
- Williams, P. G. (1998). Chaos theory tamed. In *Choice Reviews Online* (Vol. 35, Issue 08, pp. 35-4561-35-4561). <https://doi.org/10.5860/choice.35-4561>
- Zmeskal, O., Dzik, P., & Vesely, M. (2013). Entropy of fractal systems. *Computers and Mathematics with Applications*, 66(2), 135–146. <https://doi.org/10.1016/j.camwa.2013.01.017>

## Anexos

```
# -*- coding: utf-8 -*-  
"""
```

```
Created on Tue Jul 25 13:44:19 2022
```

```
@author: pedro  
"""
```

```
import numpy as np  
import scipy.io as sio  
import matplotlib.pyplot as plt  
from PIL import Image as im  
import scipy.linalg as ort
```

```
def GramSchmidt(V): #Función para ortonormalización  
    tam=V.shape  
    m=tam[0]  
    n=tam[1]  
    U=np.zeros((m,n))  
    U[0]=V[0]  
    for k in range(1,m):  
        U[k]=V[k]  
        for j in range(k):  
            U[k]=U[k]-(np.dot(V[k],U[j]))/(np.dot(U[j],U[j]))*U[j]  
  
    return ( U )
```

```
##### Solución de Sistema Henon Map #####
```

```
x0=0  
y0=0  
a=1.4#0.1#1.4 #Parámetro  
b=0.3 #Parámetro
```

```
N=int(3e4 + 1)
```

```
x=np.zeros(N)  
y=np.zeros(N)  
for i in range(N-1):  
    x[i+1]=1 - (a*x[i]**2) + y[i]  
    y[i+1]=b*x[i]
```

```
##### Armado de Contenedores en el espacio fase
```

```
epsilon=0.1#0.2 # Tamaño de Contenedor  
Alto=np.linspace(-np.ceil(np.max(y)), np.ceil(np.max(y)), int(2*np.ceil(np.max(y))/epsilon) )  
Ancho=np.linspace(-np.ceil(np.max(x)), np.ceil(np.max(x)), int(2*np.ceil(np.max(x))/epsilon) )  
bins=np.zeros((len(Ancho)-1,len(Alto)-1)) # Contenedores vacíos para almacenar las recurrencias de las trayectorias
```

```
pos=np.zeros((len(x),2)) #almacena las posiciones secuenciales
```

```
### Llenado de Contenedores con # de Recurrencias en el espacio fase
```

```
for k in range(len(x)):  
    for i in range(len(Ancho)-1):  
        for j in range(len(Alto)-1):  
            if x[k]>=Ancho[i] and x[k]<=Ancho[i+1] and y[k]>=Alto[j] and y[k]<=Alto[j+1]:  
                bins[i][j]=bins[i][j]+1  
                pos[k][0]=i  
                pos[k][1]=j
```

```
P=bins/sum(sum(bins)) #Probabilidad en cada contenedor, tal que sum(sum(P))=1
```

```
K=0
```

```
for k in range(0,N):  
    #P0=P[ int(pos[k+1][0]), int(pos[k+1][1]) ]  
    P1=P[ int(pos[k][0]), int(pos[k][1]) ]  
    K=K+( P1*np.log2(1/P1) )  
    #K=K-P1*np.log(P1)  
    #K=K+P1*np.log2(P1/P0)  
K=K/N
```

```
plt.plot(x,y,'r')  
plt.xlabel('x',fontsize=12)  
plt.ylabel('y',fontsize=12)
```

```

plt.show()
plt.imshow(np.transpose(bins))

##### Cálculo de exponentes de Lyapunov para comprobar conjetura de Kolmogorov

J=np.zeros((2,2))
W=np.identity(2) # Prepara Matriz Ortogonal

e1=0
e2=0

for i in range(0,N):
    #Analítica
    J=[ [ -2*a*x[i], 1 ], [ b,0 ] ] #Jacobiano
    #Jp=np.identity(2) + np.multiply(J,dT) #Multiplica por estalar dT
    Jp=np.identity(2) + np.multiply(J,1) #Multiplica por estalar dT
    #W=ort.orth(np.dot(Jp,W)) # Calcula la Matriz Ortogonal
    W=GramSchmidt(np.dot(Jp,W))
    e1=e1+np.log(ort.norm(W[:,0]))
    e2=e2+np.log(ort.norm(W[:,1]))
    W[:,0]=W[:,0]/ort.norm(W[:,0])
    W[:,1]=W[:,1]/ort.norm(W[:,1])

e1=e1/N
e2=e2/N
## Exponentes de Lyapunov:
L1=np.exp(e1)
L2=np.exp(e2)

print("\n")
print("Exponente de Lyapunov máximo: ")
print(e1+e2)
print("\n")
print("Exponentes de Lyapunov: ")
print([e1,e2])
print("\n Entropía K-S: ")
print(K)
print("\n")

```