Centro de Investigación en Computación

Series de Tiempo (Predicción y Caos)

Proyecto 1:

Caminante Aleatorio

Autor:

Rodrigo Gerardo Trejo Arriaga

Abril 2024

Proyecto 1:

Caminante Aleatorio

Un caminante aleatorio es un proceso estocástico que modela una secuencia de pasos tomados al azar. En el contexto de las series de tiempo, este fenómeno describe cómo una variable de interés evoluciona a lo largo del tiempo de manera que cada valor sucesivo depende del valor anterior más alguna variación aleatoria. Formalmente, se puede representar como:

$$P_t = P_{t-1} + \epsilon_t$$

donde P_t es el estado del sistema en el tiempo t, y ϵ_t es un término aleatorio que representa la innovación o perturbación en el tiempo t, usualmente asumiendo una distribución normal con media cero.

Aunque el concepto de caminante aleatorio es simple, sus aplicaciones son vastas y variadas. En economía, se utiliza para modelar la hipótesis de mercados eficientes, sugiriendo que los precios de los activos financieros siguen un camino aleatorio y, por lo tanto, son fundamentalmente impredecibles. Esta hipótesis implica que la información pasada no tiene poder predictivo sobre los movimientos futuros de los precios.

En la física, el movimiento browniano es un ejemplo clásico de un caminante aleatorio, describiendo el movimiento aparentemente aleatorio de partículas suspendidas en un fluido. Además, en ecología, los patrones de movimiento de diferentes animales pueden ser modelados como caminantes aleatorios, ayudando a entender los procesos de búsqueda de alimento y la dispersión de poblaciones.

En este proyecto modelaremos dicho proceso estocástico mediante la animación del movimiento de partículas en una cuadrícula; tomando como elementos aleatorios los siguientes parámetros:

- Posición inicial (x_0, y_0).
- Dirección de movimiento (del 1 al 8 en cuanto a pixeles).
- Distancia a recorrer en la dirección seleccionada

Dichos elementos serán generados mediante números random, los cuales seguirán distribuciones *normal*, *gamma* y *uniforme*, con el fin de observar el comportamiento del sistema ante diferentes distribuciones de probabilidad en su entrada.

Este proyecto consta de cuatro fases:

I. Generación de Números Random con Distribución Uniforme

En primer lugar, generamos un total de 100,000 números aleatorios con distribución *uniforme* que se guardaron en un archivo de texto llamado `randomsUniforme.txt`, los cuales generaron un histograma como el siguiente:

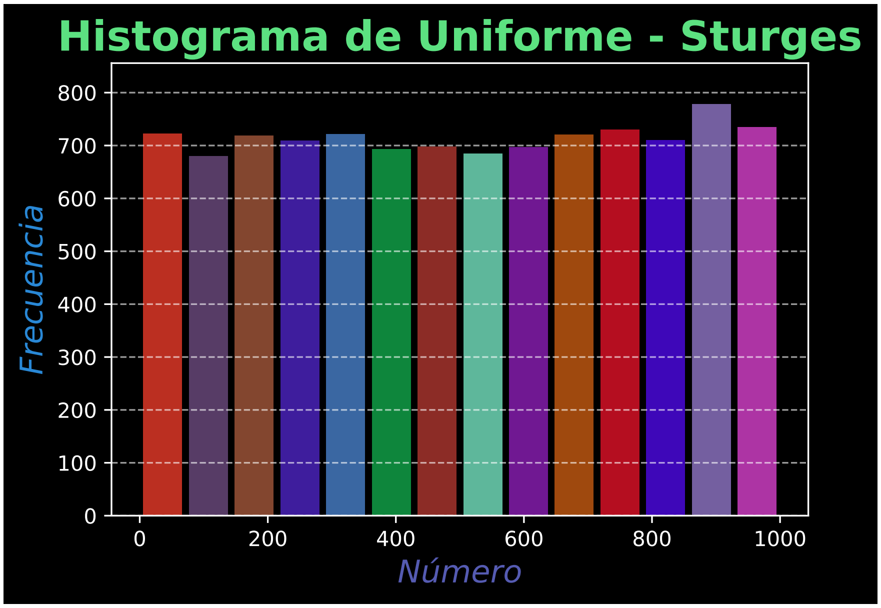


Figura 1: Histograma de números con distribución uniforme.

II. Transformación de distribución

En segundo lugar, transformamos los números generados con una distribución uniforme mediante una técnica específica. Esta transformación nos permite obtener un nuevo conjunto de números aleatorios, pero esta vez siguiendo distribuciones de probabilidad distintas, concretamente la distribución normal y la gamma.

II.1. Transformación a Distribución Normal

La transformación de variables aleatorias con distribución uniforme a una distribución normal se realiza mediante el uso de la función de error inversa (*inverse error function*), la cual juega un papel crucial en estadística para la transformación de variables aleatorias.

Dado un valor q de una distribución uniforme, el objetivo es encontrar el correspondiente valor en una distribución normal con media μ y desviación estándar σ. La transformación se lleva a cabo en dos pasos principales:

La función de error inversa se calcula utilizando una aproximación basada en una fórmula específica. Para un valor x, la función se define como:

$$\text{inverse\_erf}(x) = \text{sign}(x) \cdot \sqrt{\sqrt{\left(\frac{2}{\pi a} + \frac{\log(1-x^2)}{2}\right)^2 - \frac{\log(1-x^2)}{a}} - \left(\frac{2}{\pi a} + \frac{\log(1-x^2)}{2}\right)}$$

donde a es un coeficiente que en esta aproximación toma el valor de 0.147. Esta función permite mapear los valores de x de la distribución uniforme a valores que siguen la función de error, la cual está estrechamente relacionada con la distribución normal.

Con la función de error inversa calculada, el siguiente paso es utilizarla para transformar el cuantil q de la distribución uniforme a un valor que siga una distribución normal estándar, y luego ajustarlo a cualquier media μ y desviación estándar σ deseada. Esto se logra mediante la expresión:

$$\text{norm\_ppf}(q, \mu, \sigma) = \mu + \sigma \cdot \sqrt{2} \cdot \text{inverse\_erf}(2q - 1)$$

Este método asegura que los valores generados a partir de una distribución uniforme puedan ser transformados efectivamente a valores que siguen una distribución normal con los parámetros especificados.

Dicha transformación se aplicó a los 100,000 números distribuidos uniformemente, se guardó el resultado en un archivo `randomsNormal.txt` teniendo a la salida números normalmente distribuidos, como se muestra en el histograma:

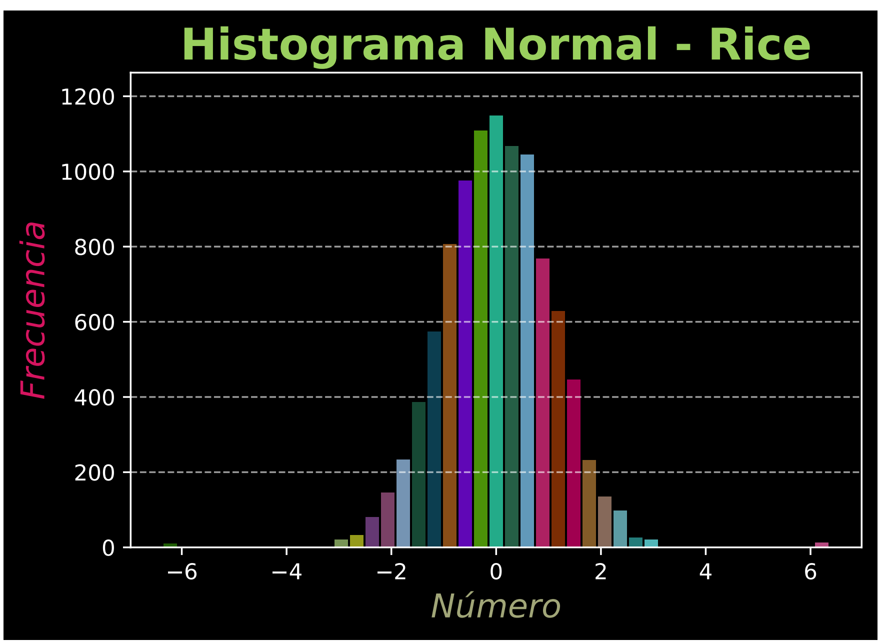


Figura 2: Histograma de números con distribución normal.

II.2. Transformación a Distribución Gamma

Para generar variables aleatorias que sigan una distribución Gamma con parámetros de forma $a > 1$ y escala θ, utilizamos el siguiente algoritmo:

Definimos $\lambda = a - 1$ y calculamos un valor c usando la expresión:

$$c = \frac{(a-1)^{a-1}e^{1-a}}{\Gamma(a)},$$

donde $\Gamma(a)$ es la función Gamma para el parámetro a.

Repetimos los siguientes pasos hasta que se acepte un candidato:

1. Generamos una variable aleatoria y de una distribución exponencial con parámetro λ, utilizando la transformación $y = -\frac{\log(1-U)}{\lambda}$, donde U es un número aleatorio uniforme entre 0 y 1.
2. Generamos un número aleatorio uniforme u.
3. Aceptamos y si $u \leq \frac{y^{a-1}e^{-y}}{c\lambda^{-a}e^{-\lambda y}}$. En caso contrario, rechazamos y y repetimos el proceso.

Una vez aceptado, el valor de salida es $y \cdot \theta$, asegurando que la variable aleatoria generada siga una distribución Gamma con los parámetros de forma a y escala θ.

De esta manera, obtenemos números con distribución gamma que se almacenaron en `randomsGamma.txt`. Su transformación la observamos en el siguiente histograma:

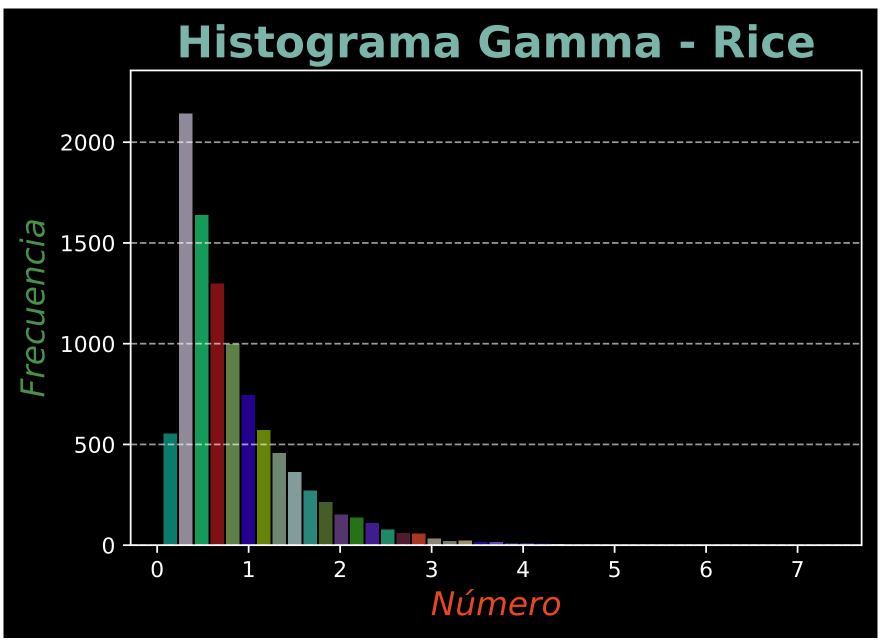


Figura 3: Histograma de números con distribución gamma.

III. Métricas de las Distribuciones de Probabilidad

Se realizó una investigación respecto a las expresiones matemáticas que definen a las distribuciones de probabilidad *uniforme, gamma y normal* (las cuales se detallan en el documento **Métricas de Probabilidad**), que se codificaron como sigue para poder aplicarlas en nuestros set de datos anteriores:

III.1. Métricas para un set de datos genérico

```
class DistribucionProbabilidad:
def __init__(self, datos):
    self.datos = np.array(datos)
    self.n = len(datos)

def media(self):
    return np.mean(self.datos)

def mediana(self):
    return np.median(self.datos)

def moda(self):
    frecuencias = {}
    for dato in self.datos:
        if dato in frecuencias:
            frecuencias[dato] += 1
        else:
            frecuencias[dato] = 1

    modas = [valor for valor, frecuencia in frecuencias.items()
        if frecuencia == max(frecuencias.values())]

    if len(modas) == 1:
        return modas[0]
    else:
        return modas

def media_geometrica(self):
    return gmean(self.datos)

def asimetria(self):
    media = self.media()
    desv_std = self.desviacion_estandar()
    suma_tercero_momento = np.sum((self.datos - media) ** 3)
    skewness = (self.n / ((self.n - 1) * (self.n - 2))) * (
        suma_tercero_momento / (desv_std ** 3))
    return skewness

def rango(self):
    return self.datos.max() - self.datos.min()

def desviacion_estandar(self):
    return np.std(self.datos, ddof=1)

```

```python
    def varianza(self):
        return np.var(self.datos, ddof=1)

    def coeficiente_variacion(self):
        return self.desviacion_estandar() / self.media()

    def percentil(self, p):
        if 0 <= p <= 100:
            return np.percentile(self.datos, p)
        else:
            raise ValueError("El percentil debe estar entre 0 y 100.")

    def cuartil(self, q):
        if q in [1, 2, 3]:
            return self.percentil(q * 25)
        else:
            np.nan

    def curtosis(self):
        media = self.media()
        s = self.desviacion_estandar()
        suma_cuarto_momento = np.sum((self.datos - media) ** 4)
        curtosis = (self.n * (self.n+1) * suma_cuarto_momento) / ((self.n-1) * (self.n-2) * (self.n-3) * s**4) - (3 * (self.n-1)**2) / ((self.n-2) * (self.n-3))
        return curtosis

    def entropia(self):
        p, counts = np.unique(self.datos, return_counts=True)
        p = counts / len(self.datos)
        entropia = -np.sum(p * np.log(p))
        return entropia

    def calcular_metricas(self, percentil, cuartil):
        resultados = {
            'Media': self.media(),
            'Mediana': self.mediana(),
            'Moda': self.moda(),
            'Media Geometrica': self.media_geometrica(),
            'Rango': self.rango(),
            'Desviacion Estandar': self.desviacion_estandar(),
            'Varianza': self.varianza(),
            f'Percentil {percentil}': self.percentil(percentil),
            f'Cuartil {cuartil}': self.cuartil(cuartil),
            'Asimetria': self.asimetria(),
            'Coeficiente de Variacion': self.coeficiente_variacion(),
            'Curtosis': self.curtosis(),
            'Entropia': self.entropia()
        }

        for metrica, valor in resultados.items():
            print(f"{metrica}: {valor}")
```

III.2. Métricas para una Distribución Uniforme

```
class DistribucionUniforme(DistribucionProbabilidad):

    def __init__(self, datos):
        super().__init__(datos)
        self.a = min(datos)
        self.b = max(datos)

    def media(self):
        return (self.a + self.b)/2

    def mediana(self):
        return (self.a + self.b)/2

    def media_geometrica(self):
        return super().media_geometrica()

    def moda(self):
        return super().moda()

    def rango(self):
        return self.b - self.a

    def varianza(self):
        return ((self.b - self.a)**2)/12

    def desviacion_estandar(self):
        return (((self.b - self.a)**2)/12) ** (1/2)

    def coeficiente_variacion(self):
        return ((((self.b - self.a)**2)/12) ** (1/2)) / ((self.
            a + self.b) / 2)

    def percentil(self, num_perc):
        p = num_perc/100
        return self.a + p*(self.b - self.a)

    def cuartil(self, num_cuartil):
        num_perc = num_cuartil * 25
        return self.percentil(num_perc)

    def asimetria(self):
        return super().asimetria()

    def curtosis(self):
        return super().curtosis()

    def entropia(self):
        return math.log(self.b - self.a)
```

```

        def calcular_metricas(self, percentil, cuartil):
            return super().calcular_metricas(percentil, cuartil)
```

III.3. Métricas para una Distribución Normal

```
    class DistribucionNormal(DistribucionProbabilidad):

        def __init__(self, datos):
            super().__init__(datos)
            self.mu = self.media()
            self.sigma = self.desviacion_estandar()

        def media(self):
            return super().media()

        def mediana(self):
            return self.mu

        def media_geometrica(self):
            return "No aplica"

        def moda(self):
            return self.mu

        def rango(self):
            return super().rango()

        def percentil(self, num_perc):
            p = num_perc/100
            if 0 <= p <= 1:
                valor_z = norm.ppf(p)
                return self.mu + valor_z * self.sigma
            else:
                return np.nan

        def cuartil(self, num_cuartil):
            if num_cuartil in [1,2,3,4]:
                c = num_cuartil * 25
                return self.percentil(c)

            else:
                return np.nan

        def varianza(self):
            return self.sigma**2

        def desviacion_estandar(self):
            return super().desviacion_estandar()

        def coeficiente_variacion(self):
            if self.mu != 0:
```

```
            return self.sigma/self.mu
        else:
            return np.nan

    def asimetria(self):
        return super().asimetria()

    def curtosis(self):
    return super().curtosis()

    def entropia(self):
        return 0.5 * np.log(2 * np.pi * np.e * self.sigma ** 2)

    def calcular_metricas(self, percentil, cuartil):
        return super().calcular_metricas(percentil, cuartil)
```

III.4. Métricas para una Distribución Gamma

```
class DistribucionGamma(DistribucionProbabilidad):

    def __init__(self, datos):
        super().__init__(datos)
        self.k, self.theta = self.estimar_params(self.datos)

    @staticmethod
    def estimar_params(data):
        def neg_log_likelihood(params, data):
            return -np.sum(gamma.logpdf(data, a=params[0],
                scale=params[1]))

        pesos_iniciales = [1.0, 1.0]
        result = minimize(neg_log_likelihood, pesos_iniciales,
            args=(data), method='L-BFGS-B', bounds=((0, None),
            (0, None)))

        k_est, theta_est = result.x
        return k_est, theta_est


    def media(self):
        return self.k * self.theta

    def mediana(self):
        return super().mediana()

    def media_geometrica(self):
        return super().media_geometrica()

    def moda(self):
        if self.k > 1:
            return (self.k - 1) * self.theta
        else:
```

```
            return np.nan

    def rango(self):
        return super().rango()

    def varianza(self):
        return self.k * self.theta**2

    def desviacion_estandar(self):
        return (self.k)**(1/2)*self.theta

    def coeficiente_variacion(self):
        return 1/(self.k)**(1/2)

    def percentil(self, num_perc):
        p = num_perc/100
        return gamma.ppf(p, self.k, scale=self.theta)

    def cuartil(self, num_cuartil):
        c = num_cuartil * 25
        return self.percentil(c)

    def asimetria(self):
        return super().asimetria()

    def curtosis(self):
        return 6/self.k

    def entropia(self):
        return self.k + np.log(self.theta) + gammaln(self.k) + (1 - self.k) * psi(self.k)

    def calcular_metricas(self, percentil, cuartil):
        return super().calcular_metricas(percentil, cuartil)
```

Así, para los sets de datos `randomsUniforme.txt`, `randomsNormal.txt` y `randomsGamma.txt` se obtuvieron los siguientes resultados:

```
Distribución Uniforme
Media: 500.0
Mediana: 500.0
Moda: 994.0
Media Geométrica: 0.0
Rango: 1000.0
Desviación Estándar: 288.67513459481285
Varianza: 83333.33333333333
Percentil 20: 200.0
Cuartil 3: 750.0
Asimetría: 0.031077114339680464
Coeficiente de Variación: 0.5773502691896257
Curtosis: -1.1631198834242384
Entropía: 6.907755278982137
```

Figura 4: Métricas para el set `randomsUniforme.txt`.

```
Distribución Normal
Media: 0.01619842591887649
Mediana: 0.01619842591887649
Moda: 0.01619842591887649
Media Geométrica: No aplica
Rango: 12.727364117383477
Desviación Estándar: 1.0437856087734922
Varianza: 1.0894883970826499
Percentil 20: -0.8622737057227252
Cuartil 3: 0.7202211204387743
Asimetría: 0.02366738300979561
Coeficiente de Variación: 64.43747151734904
Curtosis: 2.4611763333062124
Entropía: 1.461792645995829
```

Figura 5: Métricas para el set `randomsNormal.txt`.

```
Distribución Gamma
Media: 0.8564571806387282
Mediana: 0.650667238385453
Moda: 0.4486894230367014
Media Geométrica: 0.6626515448038219
Rango: 7.288659290248604
Desviación Estándar: 0.5909616096086177
Varianza: 0.3492356240312083
Percentil 20: 0.3638674608659149
Cuartil 3: 1.1492325372283594
Asimetría: 3.111463611204522
Coeficiente de Variación: 0.6900071865447968
Curtosis: 2.856659504900796
Entropía: 0.714628803071913
```

Figura 6: Métricas para el set `randomsGamma.txt`.

A continuación se analizan los resultados obtenidos al aplicar las métricas a los conjuntos de datos.

III.5. Análisis de Resultados

III.5.1. Para la distribución uniforme

Tras aplicar las métricas estadísticas a una muestra de datos generados con una distribución uniforme, hemos obtenido los siguientes resultados:

- La **media** y la **mediana** tienen el mismo valor de 500.0, lo cual es esperable en una distribución uniforme donde todos los valores tienen la misma probabilidad.
- La **moda** se presenta en 994.0. En una distribución perfectamente uniforme, la moda no está bien definida ya que cada valor es igualmente probable. Sin embargo, debido a variaciones en la muestra, se puede presentar un pico en la frecuencia.
- La **media geométrica** es 0. Este resultado sugiere que la muestra puede contener valores en o cercanos a cero, lo que arrastra el promedio geométrico hacia abajo.

- El **rango** es 1000.0, lo que indica que los datos están esparcidos uniformemente a través de un intervalo amplio.
- La **desviación estándar** y la **varianza** son coherentes con la fórmula de distribución uniforme $(b-a)^2/12$, reflejando la dispersión de los datos.
- El **coeficiente de variación** es 0.577, lo que puede considerarse bajo, indicando una baja dispersión relativa respecto a la media.
- Los valores de **percentil 20** y **cuartil 3** son consistentes con lo que se esperaría en una distribución uniforme.
- La **asimetría** se presenta con un valor de 0.031. Esto indica que la distribución de los datos es casi simétrica, con una ligera inclinación hacia la derecha.
- La **curtosis** negativa sugiere una distribución más achatada que la distribución normal.
- La **entropía** es positiva y está relacionada con el logaritmo del rango de los datos, indicando un grado de incertidumbre asociado a la variabilidad de los datos dentro del rango.

III.5.2. Para la distribución normal

- La **media**, la **mediana** y la **moda** tienen valores muy cercanos entre sí, lo que es característico de una distribución normal, que es simétrica alrededor de su media.
- El **rango** es aproximadamente 12.73, lo cual nos da una idea de la dispersión total de los datos desde el valor mínimo al máximo.
- La **desviación estándar** es de aproximadamente 1.04, indicando que la mayoría de los datos se encuentran dentro de este rango alrededor de la media en una distribución normal.
- La **varianza**, siendo el cuadrado de la desviación estándar, refleja la dispersión de los datos y en este caso es aproximadamente 1.09.
- El **percentil 20** es aproximadamente -0.86, lo que indica que el 20 % de los valores son menores que este número.
- El **cuartil 3** es aproximadamente 0.70, mostrando que el 75 % de los datos están por debajo de este valor.
- La **asimetría** tiene un valor cercano a cero (aproximadamente 0.024), lo que confirma la simetría de la distribución. Una distribución perfectamente normal tendría una asimetría de cero.
- El **coeficiente de variación** es bastante alto, aproximadamente 64.44, lo que sugiere una variabilidad relativa alta en comparación con la media.
- La **curtosis** es aproximadamente 2.46, lo cual es ligeramente superior a 3, que es la curtosis de una distribución normal estándar. Esto indica una distribución un poco más puntiaguda que la normal, con colas más pesadas.
- La **entropía**, de aproximadamente 1.46, aunque no es comúnmente discutida en el contexto de la distribución normal, proporciona una medida de incertidumbre o aleatoriedad en la distribución de los datos.

III.5.3. Para la distribución Gamma

- La **media** es aproximadamente 0.856, lo cual representa el promedio de los valores de la muestra.
- La **mediana** de 0.6506 es menor que la media, lo que indica una asimetría hacia la derecha en la distribución.
- La **moda** es aproximadamente 0.4486, lo que es esperable en una distribución Gamma asimétrica, donde la moda es menor que la media y la mediana.
- La **media geométrica** es aproximadamente 0.6626, siendo relevante en distribuciones sesgadas y generalmente menor que la media aritmética para distribuciones con asimetría positiva.
- El **rango**, de aproximadamente 7.29, muestra la diferencia entre los valores máximo y mínimo en la muestra.
- La **desviación estándar**, cercana a 0.59, indica la cantidad de variación o dispersión de los valores en la muestra.
- La **varianza** es aproximadamente 0.3492, proporcionando otra medida de la dispersión de los datos.
- El **percentil 20** es aproximadamente 0.3638, indicando que el 20 % de los valores en la distribución son inferiores a este valor.
- El **cuartil 3** es de 1.1492, lo que sugiere que el 75 % de los datos son menores que este valor.
- La **asimetría** tiene un valor alto de 3.111, confirmando el sesgo positivo de la distribución y la concentración de la mayoría de los datos hacia el lado izquierdo de la media.
- El **coeficiente de variación**, aproximadamente 0.69, muestra una variabilidad relativa en relación con la media de la muestra.
- La **curtosis**, de alrededor de 2.856, indica un pico más alto y colas más gruesas en comparación con una distribución normal (curtosis normal = 3).
- La **entropía**, de aproximadamente 0.7146, es una medida de la incertidumbre inherente en los resultados de la distribución Gamma.

III.6. Transformaciones al sistema

Con la intención de seguir caracterizando otros sets de datos y observar como se conservan los sistemas dinámicos cuando a su entrada reciben un conjunto de números aleatorios y se procesan a través de transformaciones ***lineales y no lineales*** se les aplicó a cada set de datos (`randomsUniforme.txt`, `randomsNormal.txt y randomsGamma.txt`) las siguientes funciones:

III.6.1. Transformación lineal: $x^2 - 5x + 7$

Para la transformación cuadrática con origen en datos uniformemente distribuidos:

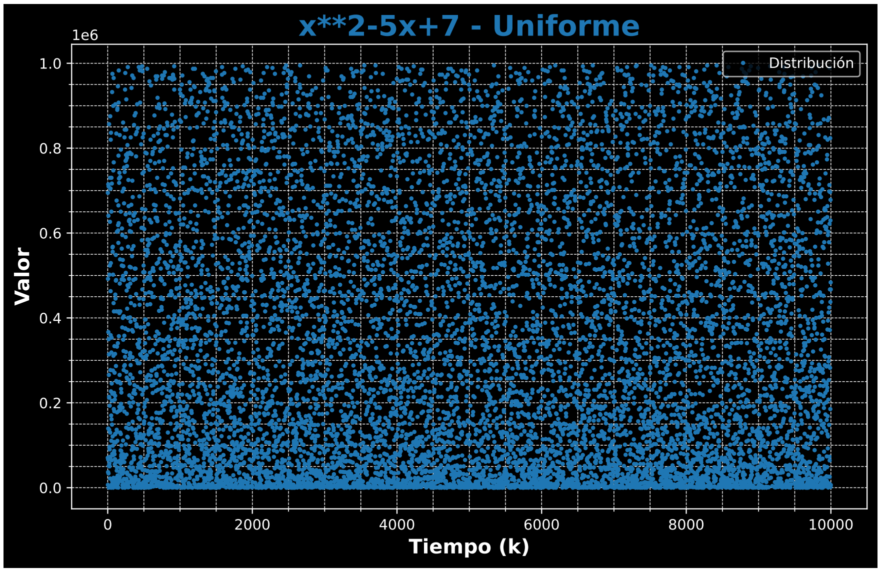


Figura 7: Gráfica de la serie de tiempo para $x^2 - 5x + 7$ con datos de distribución normal.

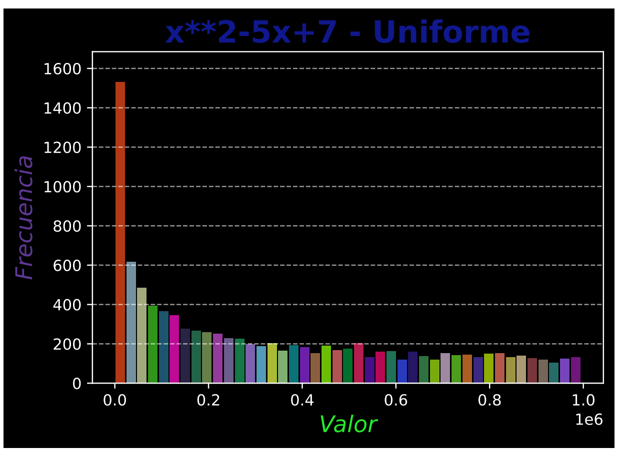


Figura 8: Histograma.

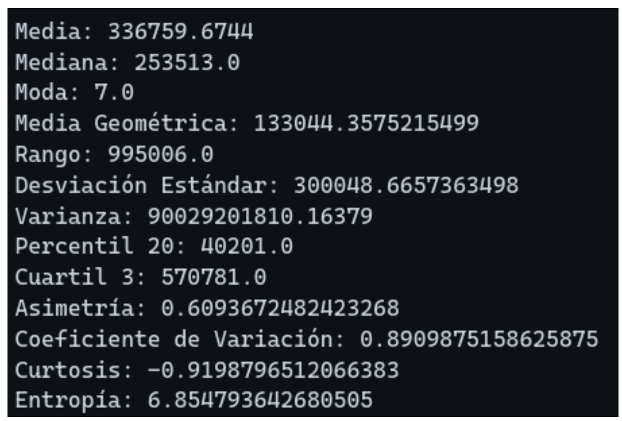

Media: 336759.6744
Mediana: 253513.0
Moda: 7.0
Media Geométrica: 133044.3575215499
Rango: 995006.0
Desviación Estándar: 300048.6657363498
Varianza: 90029201810.16379
Percentil 20: 40201.0
Cuartil 3: 570781.0
Asimetría: 0.6093672482423268
Coeficiente de Variación: 0.8909875158625875
Curtosis: -0.9198796512066383
Entropía: 6.854793642680505

Figura 9: Métricas

La serie de tiempo dada presenta una fluctuación que parece uniforme a lo largo de todos sus valores. El histograma asociado y las métricas calculadas revelan las siguientes características:

- La **media** de 336759, siendo el promedio de los datos, sugiere que la suma total de la serie de tiempo es considerablemente alta. En una distribución uniforme ideal, la media y la mediana son cercanas, lo que no se observa en este conjunto de datos.
- La **mediana** de 253513, al ser menor que la media, puede indicar que una cantidad significativa de valores de la serie de tiempo son relativamente bajos, lo que podría desplazar la media hacia arriba.

- La **moda** de 7 es atípicamente baja y específica en comparación con el rango de los datos. La presencia de una moda tan baja podría ser indicativa de un valor atípico o una acumulación de frecuencias en un punto muy específico de la distribución.
- La **media geométrica**, aproximadamente 133041, es relevante en la descripción de los datos cuando la distribución no es simétrica. Su valor inferior a la media aritmética indica una distribución sesgada positivamente.
- El **rango** de aproximadamente 995006 muestra que los datos se extienden sobre un intervalo muy amplio.
- La **desviación estándar** de 300018 es alta y sugiere que los datos están muy dispersos en relación con la media.
- El **percentil 20** de 40201 señala que hay una gran cantidad de datos que son muy bajos en comparación con el valor medio, lo cual apoya la existencia de asimetría positiva.
- El **cuartil 3** de 570781 indica que hay una amplia dispersión en los valores superiores de la serie de tiempo. En una distribución uniforme, los cuartiles estarían más equitativamente espaciados.
- La **asimetría** positiva de 0.69 sugiere una concentración de datos en el extremo inferior y una cola más larga hacia el lado derecho de la distribución, lo que es indicativo de que hay valores extremadamente altos que están alejando la media de la mediana.
- El **coeficiente de variación** de aproximadamente 0.89 indica una alta variabilidad relativa de los datos. Este valor alto sugiere una distribución con mayor dispersión en relación con la media.
- La **curtosis** de aproximadamente -0.19 es inferior a la de una distribución normal (con una curtosis de 0), lo que indica que la distribución es más plana que la normal y que tiene colas más ligeras.
- La **entropía** de 6.85 es una medida de la incertidumbre y, aunque no proporciona una indicación directa del tipo de distribución, un valor alto sugiere una mayor aleatoriedad en los datos.

Las métricas obtenidas sugieren una distribución con una asimetría positiva y un pico más plano que una distribución normal. La presencia de valores extremos es notable y aleja las características observadas de una distribución uniforme estándar. Con base a estas métricas, los datos parecen alinearse más estrechamente con una distribución que permite valores atípicos y una variabilidad considerable, tal como podría ser una distribución gamma o log-normal, dependiendo de la forma de la distribución observada en los datos.

Para la transformación cuadrática con origen en datos normalmente distribuidos:

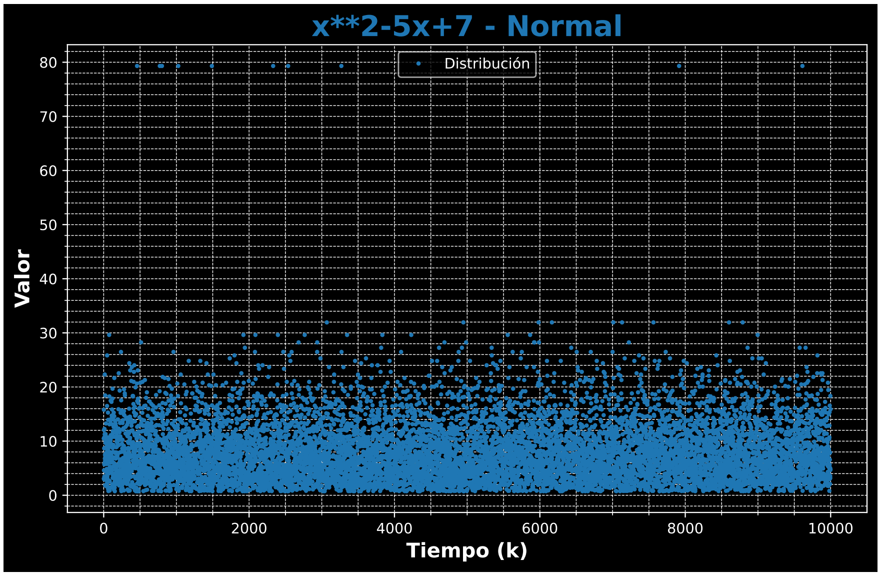


Figura 10: Gráfica de la serie de tiempo para $x^2 - 5x + 7$ con datos de distribución uniforme.

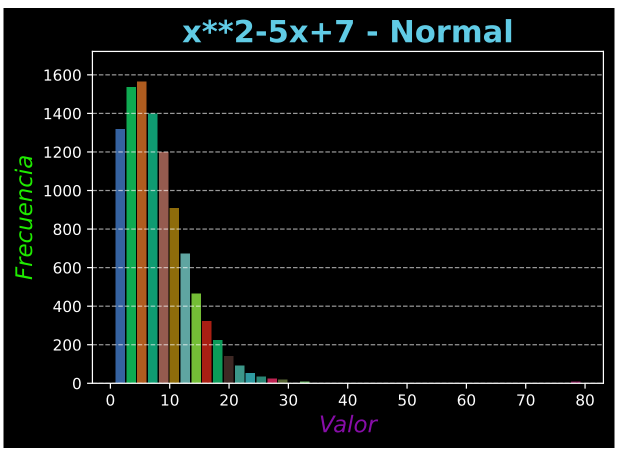
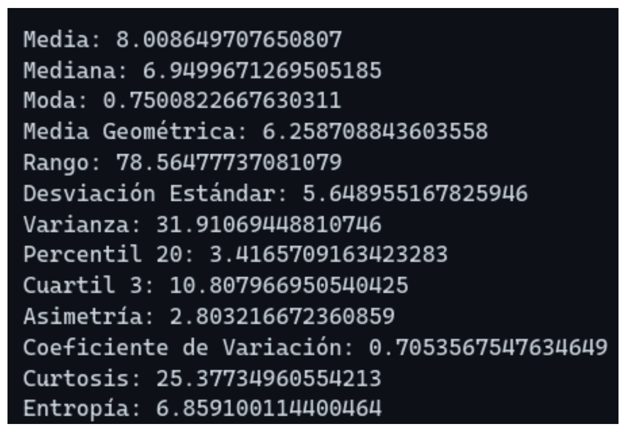


Figura 11: Histograma.



```
Media: 8.008649707650807
Mediana: 6.9499671269505185
Moda: 0.7500822667630311
Media Geométrica: 6.258708843603558
Rango: 78.56477737081079
Desviación Estándar: 5.648955167825946
Varianza: 31.91069448810746
Percentil 20: 3.4165709163423283
Cuartil 3: 10.807966950540425
Asimetría: 2.803216672360859
Coeficiente de Variación: 0.7053567547634649
Curtosis: 25.37734960554213
Entropía: 6.859100114400464
```

Figura 12: Métricas

- La **media** de aproximadamente 8.0 indica el valor promedio de la serie de tiempo. Un valor medio bajo en relación con la amplitud del rango sugiere una concentración de valores bajos.
- La **mediana** de 6.9, al estar por debajo de la media, refuerza la indicación de una asimetría positiva, donde más de la mitad de los valores están por debajo del promedio.
- La **moda** de 0.75 sugiere que el valor más frecuente es mucho más bajo que la media y la mediana, lo que podría ser indicativo de una acumulación de frecuencias cerca del límite inferior de los datos.
- La **media geométrica**, con un valor de aproximadamente 6.25, es menor que la media aritmética, lo que se alinea con una distribución que tiene una asimetría hacia la derecha.
- El **rango** es de aproximadamente 78.5, lo que indica la extensión total desde el valor más bajo hasta el más alto en la serie de tiempo.

- La **desviación estándar** es de aproximadamente 6.5, lo que muestra una variabilidad considerable alrededor de la media.
- La **varianza**, que es el cuadrado de la desviación estándar, nos dice cómo se distribuyen los valores alrededor de la media, y con un valor de aproximadamente 31.9, confirma la dispersión observada.
- El **percentil 20** de alrededor de 3.4 indica que el 20 % de los valores son inferiores a este número, lo que nos ayuda a entender la distribución de los valores más bajos.
- El **cuartil 3** de aproximadamente 10.8 indica que el 75 % de los datos se encuentran por debajo de este valor, proporcionando una medida de la distribución en el extremo superior.
- La **asimetría** con un valor de aproximadamente 2.8 sugiere una fuerte inclinación hacia la derecha, lo que significa que hay una cola más larga de valores hacia el extremo superior de la serie de tiempo.
- El **coeficiente de variación** de alrededor de 0.7 indica una variabilidad relativa moderada en relación con el valor medio.
- La **curtosis**, de aproximadamente 25.4, es mucho más alta que 0, lo que indica una distribución muy puntiaguda con colas pesadas, o la presencia de valores atípicos significativos.
- La **entropía**, con un valor de aproximadamente 6.85, refleja la aleatoriedad en la serie de tiempo. Una entropía alta sugiere una mayor incertidumbre y variabilidad en la distribución de los datos.

La combinación de una alta curtosis y una asimetría significativa sugiere que la distribución de la serie de tiempo no sigue un patrón uniforme ni normal. La presencia de una moda baja y una asimetría positiva sugiere una concentración de valores más bajos y la posibilidad de valores atípicos extremos. Estas características pueden estar asociadas con distribuciones que presentan asimetría positiva y colas pesadas, como ciertas variantes de la distribución gamma o distribuciones log-normales, pero siempre dependiendo de la naturaleza específica de los datos en estudio.

Para la transformación cuadrática con origen en datos de distribución gamma:

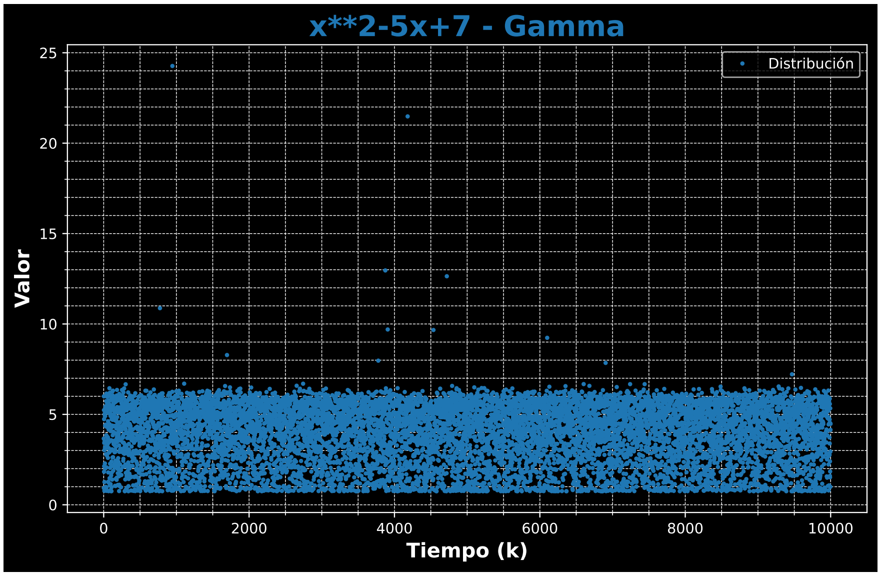


Figura 13: Gráfica de la serie de tiempo para $x^2 - 5x + 7$ con datos de distribución gamma.

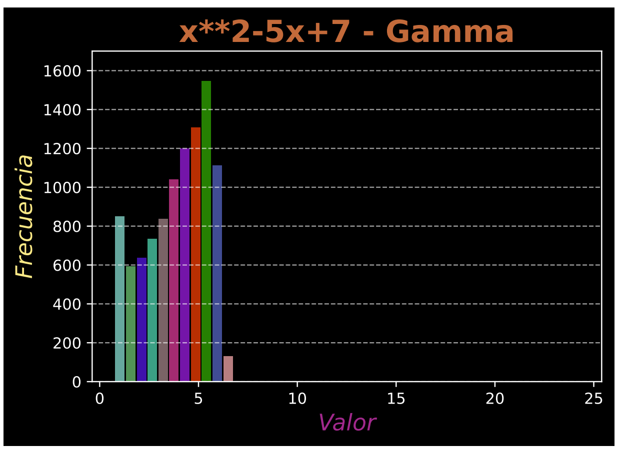
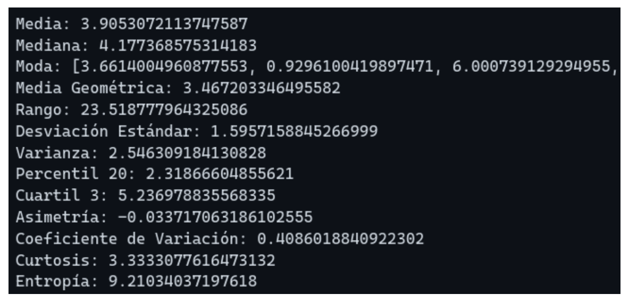


Figura 14: Histograma.



```
Media: 3.9053072113747587
Mediana: 4.177368575314183
Moda: [3.6614004960877553, 0.9296100419897471, 6.000739129294955,
Media Geométrica: 3.467203346495582
Rango: 23.518777964325086
Desviación Estándar: 1.5957158845266999
Varianza: 2.546309184130828
Percentil 20: 2.31866604855621
Cuartil 3: 5.236978835568335
Asimetría: -0.033717063186102555
Coeficiente de Variación: 0.4086018840922302
Curtosis: 3.3333077616473132
Entropía: 9.21034037197618
```

Figura 15: Métricas

- La **media** de aproximadamente 3.91 indica el promedio de los valores observados en la serie de tiempo, lo que puede sugerir una acumulación de eventos alrededor de este punto.
- La **mediana** de 4.12, siendo ligeramente mayor que la media, apunta a una distribución con una cantidad equilibrada de valores por encima y por debajo de este punto central, lo que indica cierta simetría en la distribución de los datos.
- La **moda** reporta múltiples valores (3.66, aproximadamente 0.93 y 6.00), indicando que hay varios picos de frecuencia en la distribución de los datos. Esto podría sugerir una distribución multimodal o la presencia de grupos distintos dentro de los datos.
- La **media geométrica** de 3.47 es una medida de tendencia central que es típicamente más representativa para datos que no siguen una distribución normal, especialmente cuando la distribución es sesgada.

- El **rango** de 23.58 muestra la extensión total de los datos desde el valor más bajo hasta el más alto, lo que indica una variabilidad considerable en la serie de tiempo.
- La **desviación estándar** de aproximadamente 1.60 refleja una variabilidad moderada de los datos alrededor de la media, lo que sugiere que los eventos de la serie de tiempo presentan una dispersión consistente.
- La **varianza** de aproximadamente 2.55, siendo el cuadrado de la desviación estándar, confirma la dispersión de los datos alrededor de la media.
- El **percentil 20** de 2.32 sugiere que el 20 % de los valores son menores a este número, lo que puede ser utilizado para comprender la concentración de los valores más bajos.
- El **cuartil 3** de aproximadamente 5.24 indica que el 75 % de los datos son menores que este valor, ofreciendo una visión de la distribución acumulativa hacia el extremo superior.
- Una **asimetría** ligeramente negativa de -0.03 sugiere que la serie de tiempo es casi simétrica, con una leve inclinación hacia la izquierda, lo que implica que hay una cantidad ligeramente mayor de eventos por encima de la media.
- El **coeficiente de variación** de 0.41 indica que la variabilidad de los datos es proporcionalmente moderada en relación con la media.
- Una **curtosis** de aproximadamente 3.33 sugiere una distribución con un pico similar a una distribución normal (curtosis normal = 3), lo que implica que la serie de tiempo no tiene colas inusualmente pesadas ni ligeras.
- La **entropía** de 9.21 es una medida de la incertidumbre y aleatoriedad en la serie de tiempo, con un valor más alto sugiriendo una mayor diversidad o aleatoriedad en la distribución de los datos.

En conjunto, estas métricas describen una serie de tiempo con una distribución de valores relativamente simétrica y moderadamente dispersa. La presencia de múltiples modas puede indicar la presencia de subgrupos o comportamientos distintos dentro de la serie de tiempo. La distribución de los datos no parece ser excesivamente sesgada ni puntiaguda, lo que podría ser consistente con una variedad de distribuciones de probabilidad no especificadas sin características extremas o atípicas.

III.6.2. Transformación no lineal: $5 \cdot \cos\left(\frac{x}{2}\right) + 7 \cdot \cos\left(\frac{x^2}{4}\right)$

Para la transformación no lineal con origen en datos de distribución uniforme:

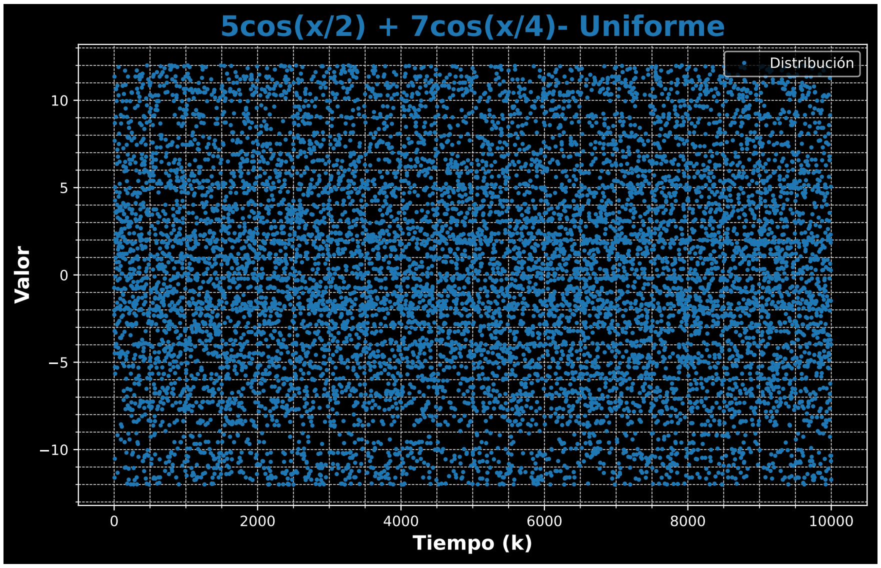


Figura 16: Gráfica de la serie de tiempo para $5 \cdot \cos\left(\frac{x}{2}\right) + 7 \cdot \cos\left(\frac{x^2}{4}\right)$ con datos de distribución uniforme.

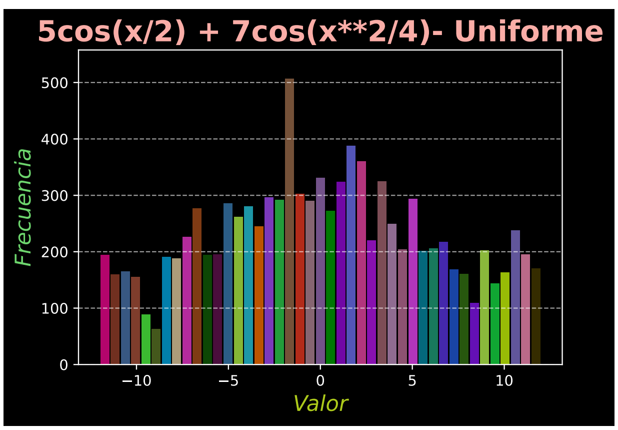
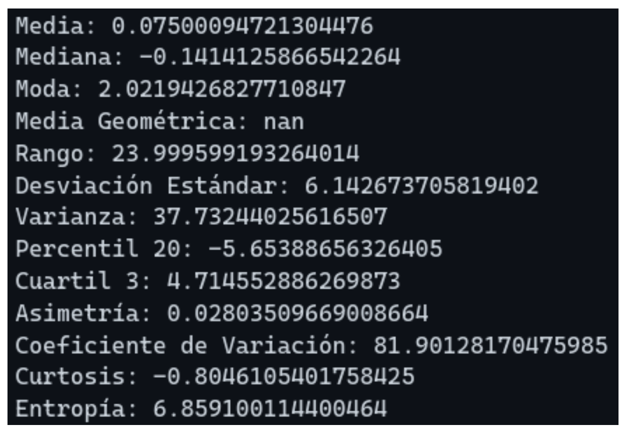


Figura 17: Histograma.



Media: 0.07500094721304476
Mediana: -0.1414125866542264
Moda: 2.0219426827710847
Media Geométrica: nan
Rango: 23.999599193264014
Desviación Estándar: 6.142673705819402
Varianza: 37.73244025616507
Percentil 20: -5.65388656326405
Cuartil 3: 4.714552886269873
Asimetría: 0.02803509669008664
Coeficiente de Variación: 81.90128170475985
Curtosis: -0.8046105401758425
Entropía: 6.859100114400464

Figura 18: Métricas

- La **media** cercana a 0.075 sugiere que el promedio de los valores oscila alrededor de cero, lo cual podría indicar una distribución de datos centrada.
- La **mediana** de -0.14, siendo ligeramente negativa, muestra que más de la mitad de los valores están por debajo de cero, lo que podría señalar una leve inclinación de los datos hacia valores negativos.
- La **moda** de aproximadamente 2.02 indica el valor que ocurre con mayor frecuencia en la serie de tiempo, sugiriendo un pico en la distribución en ese punto.
- La **media geométrica** no es aplicable (NaN), típicamente debido a la presencia de valores negativos en los datos que hacen imposible calcular esta métrica.

- El **rango** de aproximadamente 23.99 refleja una variabilidad significativa en los datos desde el valor más bajo al más alto.
- La **desviación estándar** de alrededor de 6.14 indica que los valores de la serie de tiempo están dispersos de manera considerable respecto a la media.
- La **varianza** de aproximadamente 37.73 cuantifica la dispersión observada y es consistente con una desviación estándar elevada.
- El **percentil 20** de -5.54 muestra que el 20 % de los valores están concentrados hacia el extremo inferior del rango de la serie de tiempo.
- El **cuartil 3** de aproximadamente 4.71 indica que el 75 % de los valores son inferiores a este número, proporcionando una perspectiva de la distribución hacia el extremo superior.
- Una **asimetría** de aproximadamente 0.02 sugiere que la serie de tiempo es casi simétrica con una muy ligera inclinación hacia la derecha.
- El **coeficiente de variación** de alrededor de 81.90 es extremadamente alto y destaca una variabilidad relativa excepcionalmente grande en relación con la media, lo que puede indicar una amplia gama de comportamientos o estados dentro de la serie de tiempo.
- Una **curtosis** de aproximadamente -0.84 es menor que cero, lo que indica una distribución más aplanada que la normal y con colas más ligeras.
- La **entropía** de 6.85 muestra la incertidumbre y la aleatoriedad en la distribución de la serie de tiempo, siendo un valor alto que sugiere una gran diversidad en los datos.

Este análisis muestra una serie de tiempo con una distribución de valores que tiene una gran variabilidad y una distribución aplanada. La simetría casi perfecta y la alta variabilidad relativa podrían ser características de una serie de tiempo con influencias aleatorias significativas y sin un patrón o tendencia dominante.

Para la transformación no lineal con origen en datos de distribución normal:

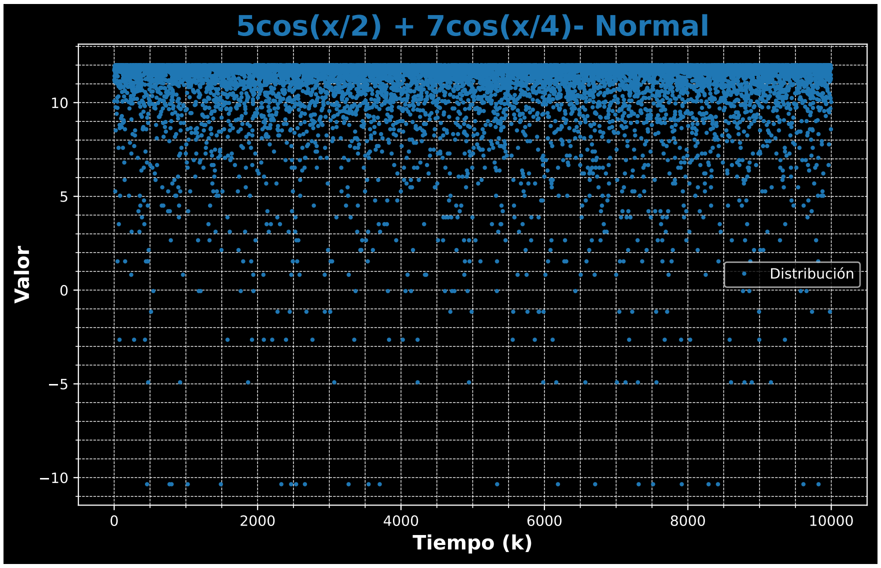


Figura 19: Gráfica de la serie de tiempo para $5 \cdot \cos\left(\frac{x}{2}\right) + 7 \cdot \cos\left(\frac{x^2}{4}\right)$ con datos de distribución normal.

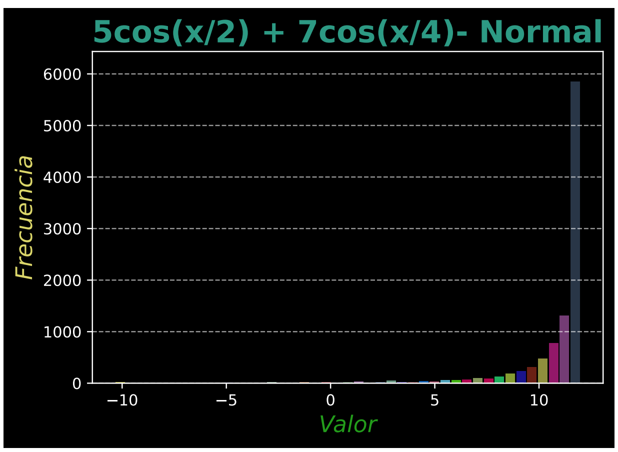
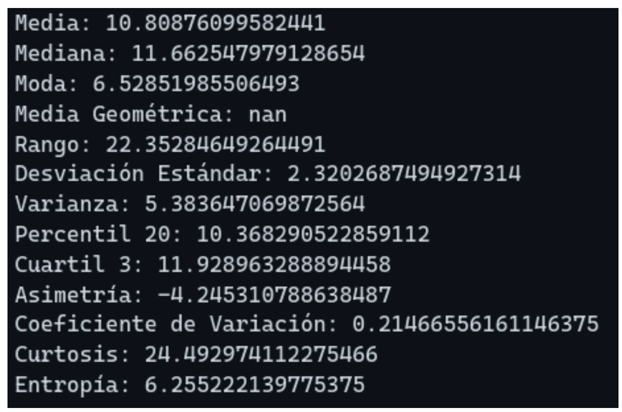


Figura 20: Histograma.



```
Media: 10.80876099582441
Mediana: 11.662547979128654
Moda: 6.52851985506493
Media Geométrica: nan
Rango: 22.35284649264491
Desviación Estándar: 2.3202687494927314
Varianza: 5.383647069872564
Percentil 20: 10.368290522859112
Cuartil 3: 11.928963288894458
Asimetría: -4.245310788638487
Coeficiente de Variación: 0.21466556161146375
Curtosis: 24.492974112275466
Entropía: 6.255222139775375
```

Figura 21: Métricas

- La **media** de aproximadamente 10.81 indica el valor promedio de la serie de tiempo, lo cual sugiere una tendencia central hacia el extremo superior del rango de los datos.
- La **mediana** de 11.66, ligeramente superior a la media, puede señalar una ligera asimetría en los datos, con una concentración de valores más altos.
- La **moda** de aproximadamente 5.63 representa el valor más frecuente dentro de la serie de tiempo y parece estar significativamente desplazada de la media y la mediana, lo que indica una distribución de los valores potencialmente bimodal o multimodal.
- La **media geométrica** no se puede calcular (NaN) debido a la presencia de valores negativos en los datos.

- El **rango** de alrededor de 22.35 muestra una variabilidad significativa entre los valores máximos y mínimos observados en la serie de tiempo.
- La **desviación estándar** de aproximadamente 2.32 refleja la cantidad de dispersión que existe alrededor de la media de los datos.
- La **varianza** de alrededor de 5.38 cuantifica la dispersión de los datos y, junto con la desviación estándar, confirma la presencia de variabilidad en la serie de tiempo.
- El **percentil 20** de 10.36 indica que el 20 % de los datos son menores o iguales a este valor, proporcionando información sobre la distribución acumulativa de los valores inferiores.
- El **cuartil 3** de 11.93 refleja el valor por debajo del cual se encuentra el 75 % de los datos, lo que ayuda a entender la concentración de los valores superiores.
- Una **asimetría** de -4.24 muestra un sesgo considerable hacia la izquierda, indicando una concentración de valores más hacia el lado derecho de la media.
- El **coeficiente de variación** de 0.21 sugiere que la variabilidad de los datos es baja en comparación con la media, lo que indica una relativa uniformidad en la dispersión de los datos.
- La **curtosis** extremadamente alta de 24.49 sugiere una distribución muy puntiaguda con colas pesadas, lo que puede implicar la presencia de valores atípicos o un agrupamiento muy denso alrededor de la media.
- La **entropía** de 6.25 refleja la incertidumbre y la aleatoriedad de los datos. Un valor más alto indica una mayor diversidad o imprevisibilidad en la distribución de la serie de tiempo.

Estas métricas indican una serie de tiempo con un marcado pico central y una notable simetría hacia la derecha, aunque la asimetría negativa y la curtosis alta sugieren una complejidad subyacente. Estos hallazgos son típicos de series que contienen eventos o comportamientos extremos y valores que se aglomeran en torno a un punto central.

Para la transformación no lineal con origen en datos de distribución gamma:

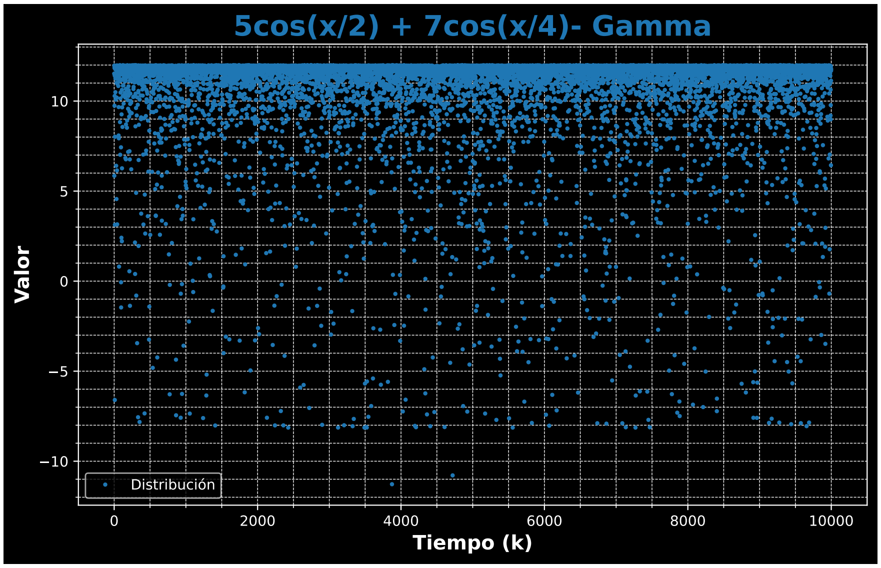


Figura 22: Gráfica de la serie de tiempo para $5 \cdot \cos\left(\frac{x}{2}\right) + 7 \cdot \cos\left(\frac{x^2}{4}\right)$ con datos de distribución gamma.

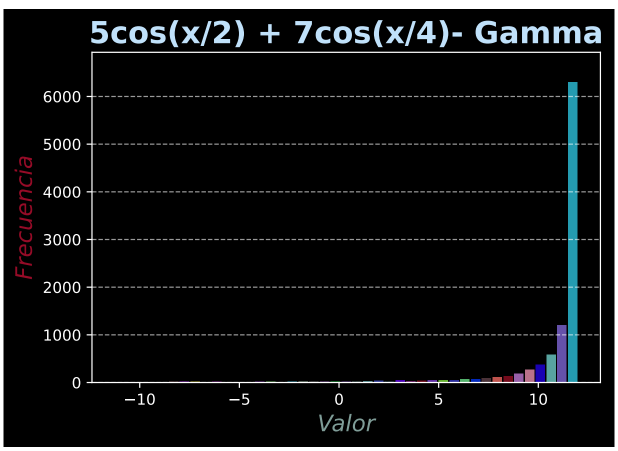


Figura 23: Histograma.

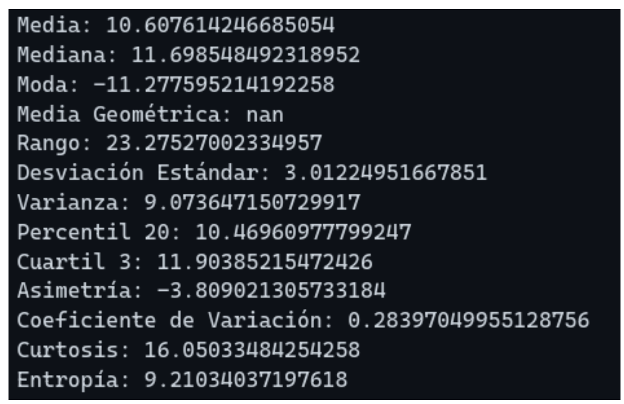

Media: 10.607614246685054
Mediana: 11.698548492318952
Moda: -11.277595214192258
Media Geométrica: nan
Rango: 23.27527002334957
Desviación Estándar: 3.01224951667851
Varianza: 9.073647150729917
Percentil 20: 10.46960977799247
Cuartil 3: 11.90385215472426
Asimetría: -3.809021305733184
Coeficiente de Variación: 0.28397049955128756
Curtosis: 16.05033484254258
Entropía: 9.21034037197618

Figura 24: Métricas

- La **media** de aproximadamente 10.61 puede sugerir que la distribución de los valores tiende a ser mayor en comparación con el punto de equilibrio de cero, indicando una posible acumulación de eventos en el lado positivo.
- La **mediana** de 11.69, mayor que la media, indica que más de la mitad de los valores están por encima del promedio, reforzando la idea de que hay una tendencia hacia valores mayores en la serie de tiempo.
- La **moda** de -11.28 es inusualmente baja y distinta de la media y la mediana, lo cual es característico de una distribución que podría ser bimodal o tener un comportamiento atípico en su distribución de valores.

- La **media geométrica** no se calcula (NaN), lo que comúnmente ocurre debido a la presencia de valores negativos en los datos, lo que impide el cálculo de esta métrica.
- El **rango** de 23.25 muestra la diferencia entre los valores máximo y mínimo en la serie de tiempo, lo que indica una extensa variabilidad de los datos.
- La **desviación estándar** de aproximadamente 3.01 sugiere una dispersión considerable de los datos alrededor de la media, lo que indica una amplia gama de comportamientos o valores en la serie de tiempo.
- La **varianza** de aproximadamente 9.07, como el cuadrado de la desviación estándar, cuantifica la dispersión de los datos y es consistente con una desviación estándar alta.
- El **percentil 20** de 10.47 muestra que el 20 % de los datos son menores o iguales a este valor, proporcionando una indicación de la distribución acumulativa de los valores inferiores.
- El **cuartil 3** de 11.90 indica que el 75 % de los valores están por debajo de este número, lo que ayuda a entender la concentración de los valores superiores en la serie de tiempo.
- Una **asimetría** de -3.89 sugiere un sesgo significativo hacia la izquierda, lo que indica una cola larga de eventos hacia el lado izquierdo de la distribución.
- El **coeficiente de variación** de 0.28 sugiere una baja variabilidad relativa de los datos en comparación con la media, lo que podría indicar una concentración de valores alrededor de la media.
- La **curtosis** extremadamente alta de 16.05 indica una distribución con un pico muy agudo y colas pesadas, lo que puede sugerir la presencia de valores extremos o comportamientos muy concentrados alrededor de la media.
- La **entropía** de 2.91 refleja la incertidumbre y la diversidad en la distribución de la serie de tiempo, con un valor más alto indicando una mayor imprevisibilidad o variedad en los datos.

Estos resultados nos llevan a concluir que estamos observando una serie de tiempo con una distribución de datos que tiene una asimetría significativa hacia la izquierda y una curtosis muy elevada. Estas características sugieren una serie con una gran cantidad de valores concentrados alrededor de un punto central y una posible presencia de valores atípicos que influyen en la forma de la distribución.

IV. Caminante Aleatorio

Con base en los datos generados con una distribución uniforme (`randomsUniforme.txt`), distribución normal (`randomsNormal.txt`) y distribución gamma (`randomsGamma.txt`) se plantea crear la animación del caminante aleatorio con el fin de observar su comportamiento al ser alimentado con números aleatorios basados en estas de probabilidad, y analizar las series de tiempo (mediante las métricas) generadas por:

- Dirección tomada.
- Distancia recorrida en cada paso.
- Choques con pared
- Posición (x,y) en el plano a lo largo del tiempo.

IV.1. Procedimiento

Se crea una cuadrícula en la que se posicionan n caminantes aleatorios (cuyo número fue definido por el usuario) en una posición aleatoria.

Luego, se crea una instancia para cada caminante, cargando el archivo de la distribución de probabilidad seleccionada.

En tercer lugar, se elige un número random que representará la dirección en la que se moverá el caminante, como se muestra en la figura 25. Posteriormente se elige otro número random que representará la distancia que el caminante recorrerá en la dirección ya seleccionada.

A su vez, por cada movimiento, se registrán los choques que se dan con la pared, cuyo índice es el que se muestra en la figura 25. Así como su posición x,y en el mapa.

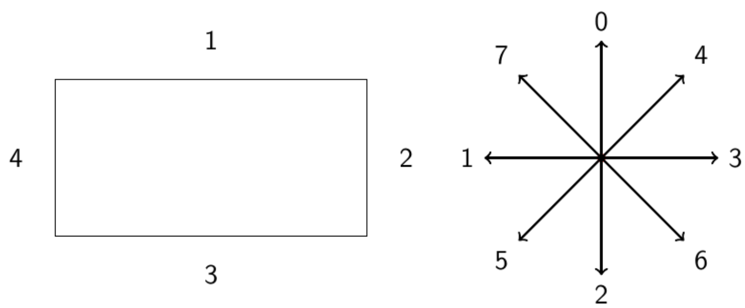


Figura 25: Índices de las paredes y las direcciones.

Nota: Por simplicidad, aunque se observen más caminantes aleatorios en cada animación, se analizará solamente al de color ***blanco***.

IV.2. Caminante Aleatorio con Distribución Uniforme

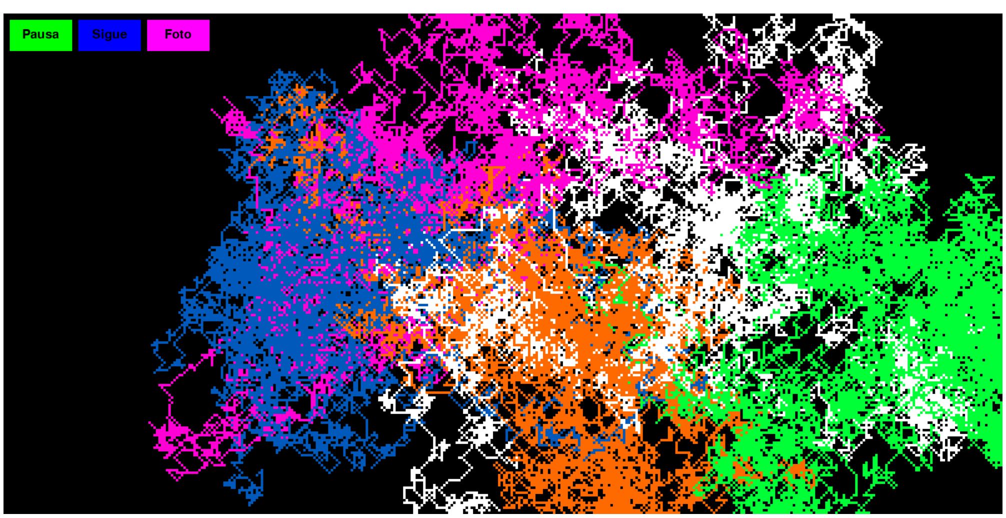


Figura 26: Caminante Aleatorio - Distribución Uniforme

IV.2.1. Direcciones

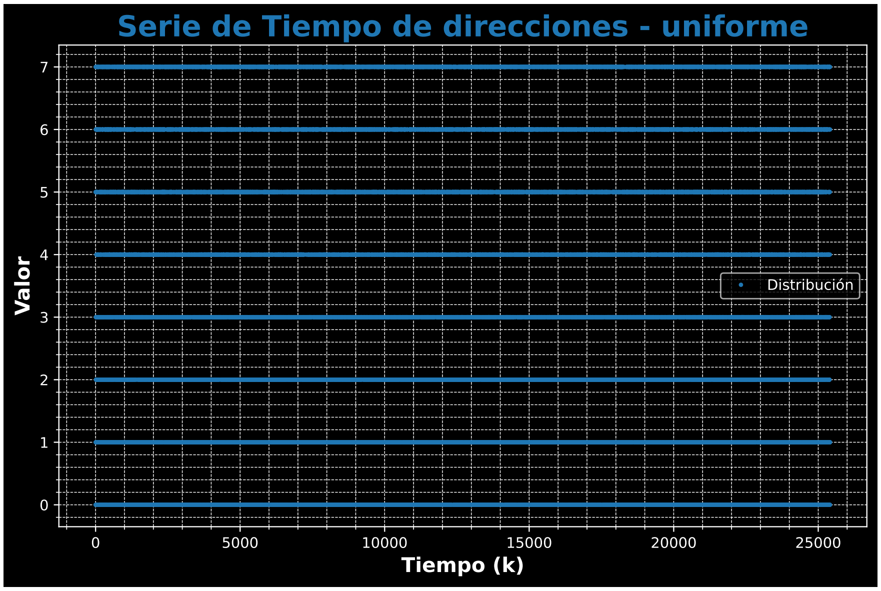


Figura 27: Gráfica de la serie de tiempo para las direcciones del caminante con distribución uniforme.

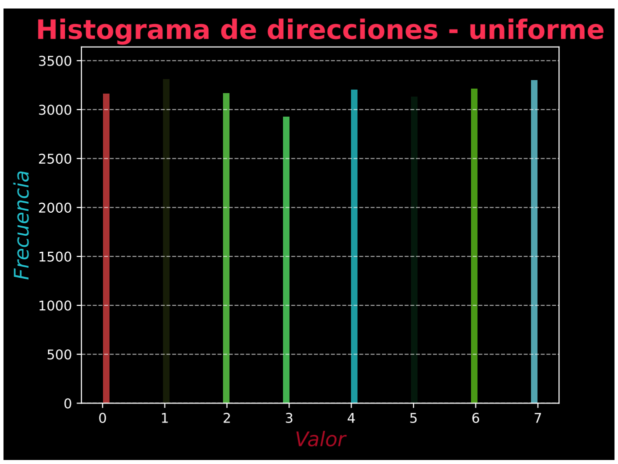


Figura 28: Histograma.

```
Media: 3.51282858492051
Mediana: 4.0
Moda: 1
Media Geométrica: 0.0
Rango: 7
Desviación Estándar: 2.310129305869035
Varianza: 5.336697409834949
Percentil 20: 1.0
Cuartil 3: 6.0
Asimetría: -0.0029643161914408604
Coeficiente de Variación: 0.6576265394177523
Curtosis: -1.2637414575100387
Entropía: 2.07881609358905
```

Figura 29: Métricas

- La **mediana** de 4.0, que es el valor central de los datos, sugiere que la mitad de las distancias son mayores que 4.0 y la otra mitad son menores, lo que indica un equilibrio en la distribución de las direcciones.
- La **moda** de 1 sugiere que el valor más comúnmente observado para la distancia en la serie de tiempo es 1, lo que podría indicar una frecuencia alta de este resultado específico o un posible punto de concentración en los datos.
- La **media geométrica** es 0.0, lo cual generalmente significa que hay valores de cero en los datos, afectando este promedio.
- El **rango** de 7 muestra la extensión completa de los datos desde el valor más bajo hasta el más alto registrado.

- Una **asimetría** de aproximadamente -0.0029 sugiere que la distribución es muy simétrica, con una ligera inclinación hacia la izquierda.
- El **coeficiente de variación** de 0.6576 refleja una variabilidad relativa moderada en comparación con la media, lo que podría indicar un grado de homogeneidad en las distancias registradas.
- La **curtosis** de -1.25, es típica de la distribución uniforme.
- La **entropía** de 2.07, similar al de aplicar el $ln(b-a) = ln(7)$, que es la métrica de la entropía en la distribución uniforme.

El análisis de estas métricas sugiere una serie de tiempo de distancias con una distribución bastante simétrica y un grado de variabilidad moderado. El perfil aplanado de la distribución, junto con la simetría y el rango de los datos, puede indicar una distribución **uniforme** de las distancias.

IV.2.2. Distancias

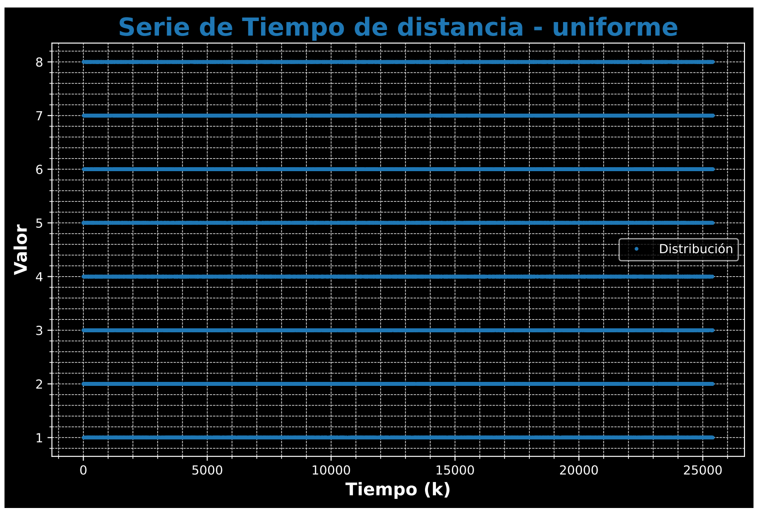


Figura 30: Gráfica de la serie de tiempo para las distancias del caminante con distribución uniforme.

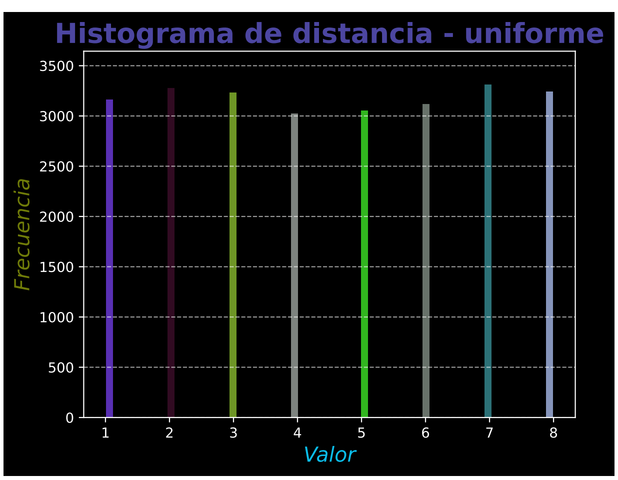


Figura 31: Histograma.

```
Media: 4.508381866834567
Mediana: 5.0
Moda: 7
Media Geométrica: 3.7652769716737735
Rango: 7
Desviación Estándar: 2.308121683936398
Varianza: 5.327425707857393
Percentil 20: 2.0
Cuartil 3: 7.0
Asimetría: 0.0007205614186959583
Coeficiente de Variación: 0.5119623297475865
Curtosis: -1.2664414722885056
Entropía: 2.078954443700381
```

Figura 32: Métricas

- La **mediana** de 5.0, que coincide con el valor medio del rango, indica una distribución simétrica de los valores en torno a este punto central.
- El **rango** de 7 sugiere que los valores de la serie de tiempo se extienden uniformemente a lo largo del espectro de datos posible.
- La **desviación estándar** de aproximadamente 2.30, junto con la varianza de 5.32, señalan una dispersión moderada de los valores alrededor de la media.
- El **percentil 20** y el **cuartil 3**, ambos con valores de 2.0 y 7.0 respectivamente, refuerzan la idea de una distribución uniforme, donde los valores están igualmente dispersos a lo largo del intervalo.
- La **asimetría** cercana a cero sugiere que la serie de tiempo es casi perfectamente simétrica.
- El **coeficiente de variación** de 0.51 indica una variabilidad relativa moderada con respecto a la media de la serie de tiempo.
- Una **curtosis** negativa de -1.26 es típica de la distribución uniforme.
- La **entropía** de 2.07, que es relativamente baja, sugiere que hay un menor grado de incertidumbre o aleatoriedad en la distribución de los datos y es similar al de aplicar el $ln(b-a) = ln(8-1) = ln(7)$, que es la métrica de la entropía en la distribución uniforme

Las métricas proporcionadas son características de una distribución que se distribuye de manera bastante **uniforme** a lo largo de su rango, con una varianza moderada y sin sesgo aparente en ninguna dirección.

IV.2.3. Posiciones

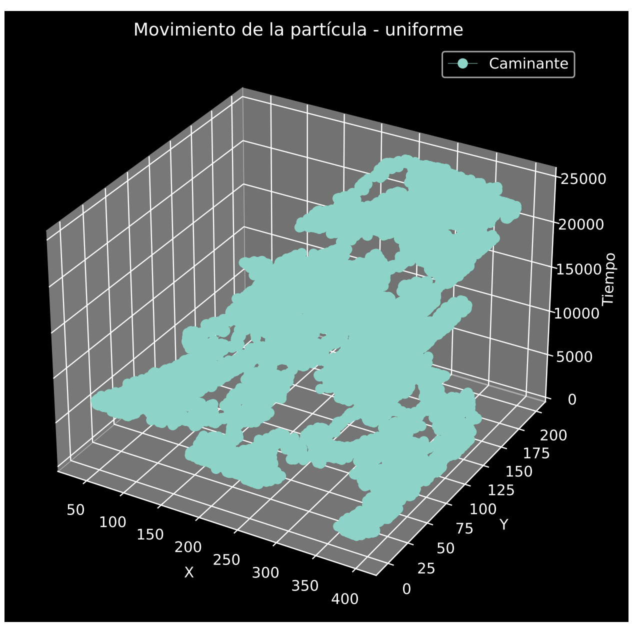


Figura 33: Gráfica de la serie de tiempo para las posiciones del caminante con distribución uniforme.

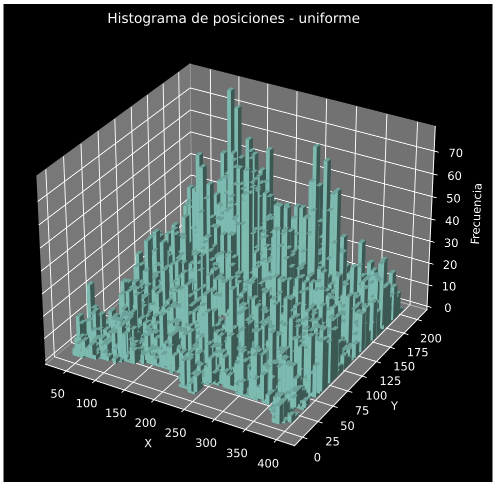
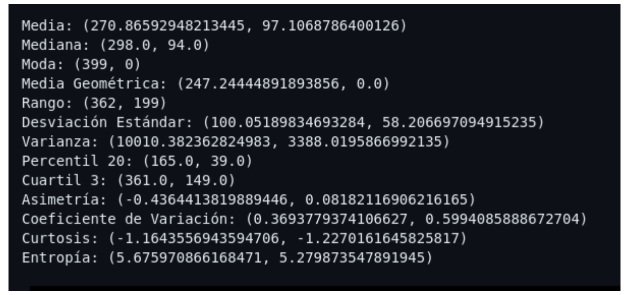


Figura 34: Histograma.



```
Media: (270.86592948213445, 97.1068786400126)
Mediana: (298.0, 94.0)
Moda: (399, 0)
Media Geométrica: (247.24444891893856, 0.0)
Rango: (362, 199)
Desviación Estándar: (100.05189834693284, 58.206697094915235)
Varianza: (10010.382362824983, 3388.0195866992135)
Percentil 20: (165.0, 39.0)
Cuartil 3: (361.0, 149.0)
Asimetría: (-0.4364413819889446, 0.08182116906216165)
Coeficiente de Variación: (0.3693779374106627, 0.5994085888672704)
Curtosis: (-1.1643556943594706, -1.2270161645825817)
Entropía: (5.675970866168471, 5.279873547891945)
```

Figura 35: Métricas

- La **media** en las coordenadas (270.86, 94.12), sugiere que el centro de masa de la distribución de posiciones está situado en esta localización en el espacio bidimensional.
- La **mediana** en las coordenadas (298.0, 94.0), muy cercana a la media, indica que la mitad de las posiciones caen por debajo de este valor y la otra mitad por encima, lo que apunta a una distribución equitativa en ambas direcciones del plano.
- El **rango**, con valores máximos en (362, 189), muestra la extensión total de los datos en el espacio bidimensional.
- La **desviación estándar**, con valores aproximados de (106.81, 56.20), refleja la dispersión de las posiciones respecto a la media en cada dirección.
- Una **asimetría** negativa de aproximadamente -4.30 en la dirección 'x' y positiva de 0.08 en la dirección 'y' sugiere una cola más pesada hacia el lado izquierdo en la dirección 'x' y una distribución más simétrica en la dirección 'y'.
- El **coeficiente de variación** de aproximadamente 0.39 en 'x' y 0.59 en 'y', indica una variabilidad relativa menor en la dirección 'x' en comparación con 'y'.
- La **curtosis**, con un valor de 1.65 en 'x' y 28.37 en 'y', sugiere una distribución más plana de lo normal en 'x' y una distribución muy puntiaguda con colas pesadas en 'y'.
- La **entropía** de 6.57 en 'x' y 2.52 en 'y', mide el grado de incertidumbre y aleatoriedad en la distribución de las posiciones en ambas direcciones, siendo relativamente alta en 'x' e indicando más uniformidad en 'y'.

Los valores indican una distribución con una dispersión considerable y una asimetría notable en la dirección 'x', mientras que en la dirección 'y', los valores sugieren una mayor concentración alrededor de la moda con colas pesadas. Estas características podrían alinearse con una serie de tiempo que contiene un área de actividad concentrada y comportamientos atípicos en determinadas regiones del espacio bidimensional analizado.

IV.2.4. Choques contra paredes

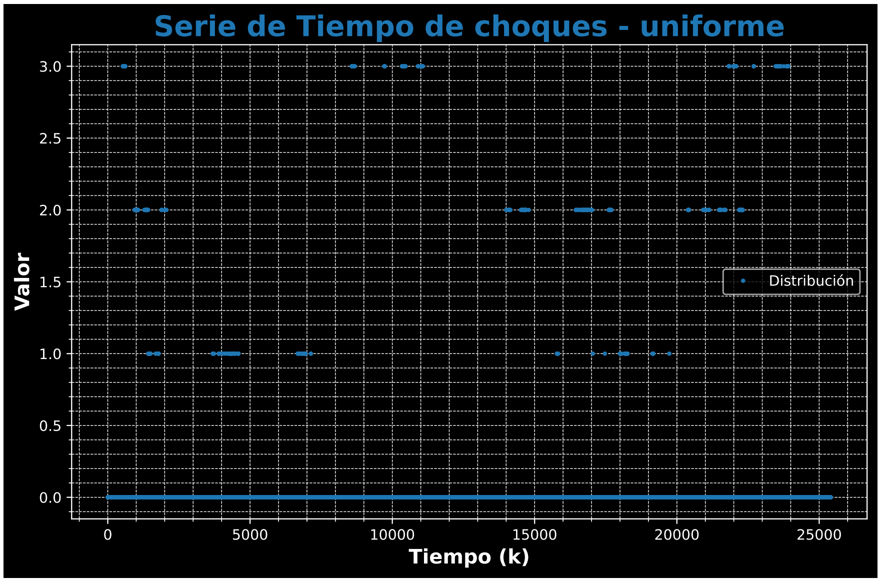


Figura 36: Gráfica de la serie de tiempo para los choques del caminante con distribución normal.

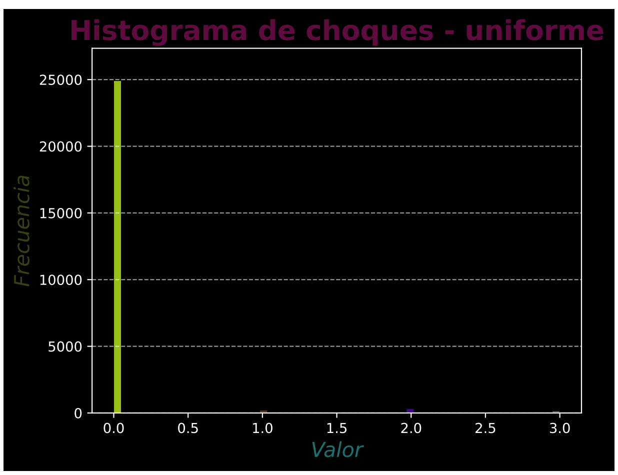
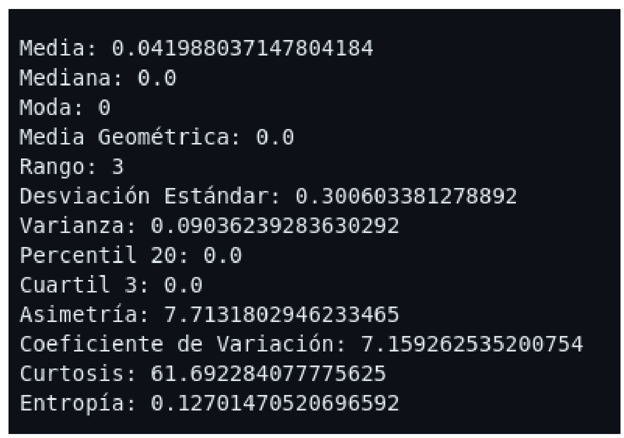


Figura 37: Histograma.



```
Media: 0.041988037147804184
Mediana: 0.0
Moda: 0
Media Geométrica: 0.0
Rango: 3
Desviación Estándar: 0.300603381278892
Varianza: 0.09036239283630292
Percentil 20: 0.0
Cuartil 3: 0.0
Asimetría: 7.7131802946233465
Coeficiente de Variación: 7.159262535200754
Curtosis: 61.692284077775625
Entropía: 0.12701470520696592
```

Figura 38: Métricas

- La **mediana** y la **moda** de 0 sugieren que la mayor parte de los valores en la serie de tiempo son ceros, lo que puede indicar ausencia de choques durante la mayoría de las observaciones.
- La **media geométrica** es 0, lo que confirma que los valores son predominantemente ceros o valores cercanos a cero.
- La **desviación estándar** de 0.3006 es relativamente pequeña, lo que indica que hay poca variación en los valores y que la mayoría de los valores están cerca de la media.
- El **percentil 20** de 0 refuerza el hallazgo de que al menos el 20 % de los datos son ceros.
- El **cuartil 3** de 0 sugiere que hasta el 75 % de los datos podrían ser ceros, lo que refleja la alta frecuencia de valores sin choques en la serie de tiempo.

- Una **asimetría** de aproximadamente 7.71 indica una distribución muy sesgada, con una cola hacia valores mayores que la mayoría de las observaciones.
- El **coeficiente de variación** de aproximadamente 7.15, al ser relativamente alto, sugiere que la variabilidad de los datos es significativa en comparación con la media, lo cual es notable dado que la media es muy baja.
- La **curtosis** extremadamente alta de aproximadamente 61.69 sugiere una distribución con un pico muy pronunciado en cero y colas muy gruesas, lo que indica una concentración significativa de ceros y la presencia de valores atípicos ocasionales.
- La **entropía** de 0.1270 es baja, lo que implica una alta predecibilidad en la serie de tiempo, probablemente debido a la gran cantidad de ceros.

Dado el predominio de ceros y la baja variabilidad general con la excepción de algunos valores atípicos, la serie de tiempo presenta características de una distribución donde la mayoría de los eventos no registran un cambio, pero ocasionalmente se observan picos significativos.

IV.3. Caminante Aleatorio con Distribución Normal

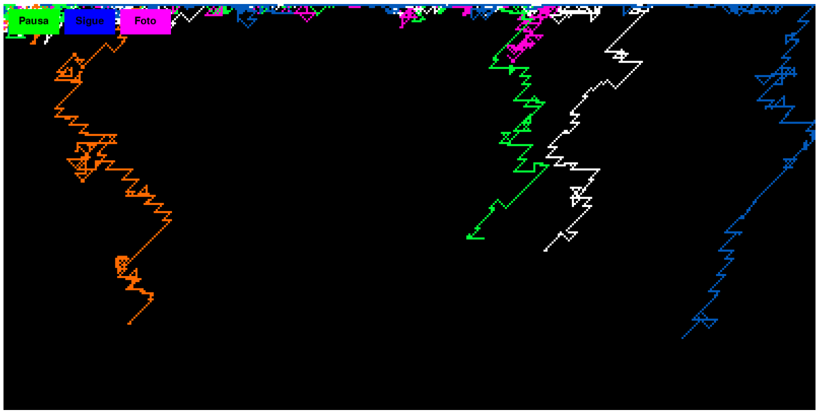


Figura 39: Caminante Aleatorio - Distribución Normal

IV.4. Direcciones

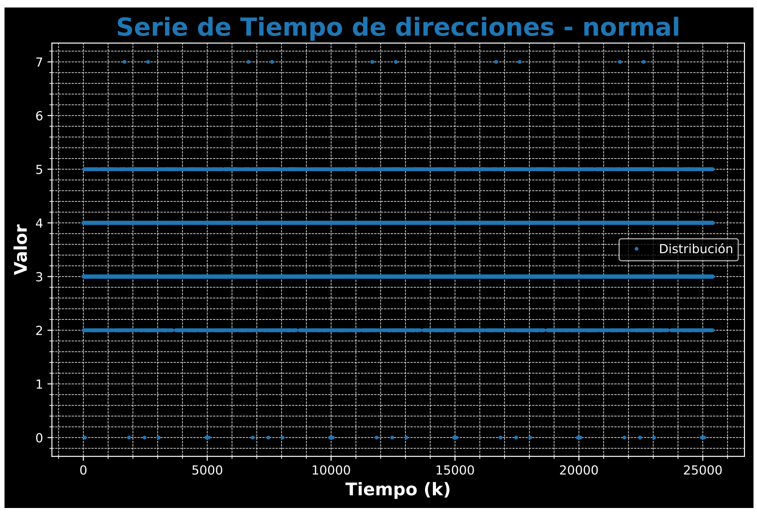


Figura 40: Gráfica de la serie de tiempo para las direcciones del caminante con distribución normal.

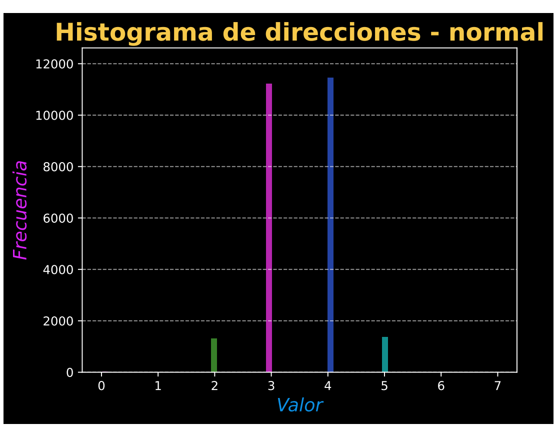


Figura 41: Histograma.

```
Media: 3.5058240201479616
Mediana: 4.0
Moda: 4
Media Geométrica: 0.0
Rango: 7
Desviación Estándar: 0.6919921788531713
Varianza: 0.4788531755939594
Percentil 20: 3.0
Cuartil 3: 4.0
Asimetría: -0.08213776900766163
Coeficiente de Variación: 0.19738360364818486
Curtosis: 0.5118522551036051
Entropía: 1.0414062357097533
```

Figura 42: Métricas

- La **media** de aproximadamente 3.51 sugiere que el promedio de los valores observados se inclina ligeramente hacia el extremo inferior del rango.
- La **mediana** de 4.0 indica que la mitad de los valores son menores que 4 y la otra mitad mayores, apuntando a una distribución equilibrada alrededor de este punto.
- La **moda** de 4 señala que el valor más frecuente en la serie de tiempo es 4, lo cual puede indicar una concentración de medidas alrededor de este valor.
- El **rango** de 7 muestra la extensión completa desde el valor más bajo al más alto.
- La **desviación estándar** de aproximadamente 0.6199 refleja que hay poca dispersión de los valores respecto a la media.
- La **varianza** de 0.4788 es una medida de la dispersión que confirma la baja variabilidad de los datos alrededor de la media.
- La **asimetría** cercana a cero implica que la serie de tiempo tiene una distribución simétrica.
- El **coeficiente de variación** de aproximadamente 0.19 indica una baja variabilidad relativa en relación con la media de la serie de tiempo.
- Una **curtosis** negativa de -0.081 sugiere que la serie de tiempo tiene una distribución menos concentrada que la normal, con colas más ligeras.
- La **entropía** muy baja de 0.141 implica que hay poca incertidumbre o aleatoriedad en la serie de tiempo, posiblemente debido a una alta frecuencia de valores repetidos o un rango limitado de valores únicos. Además es similar a la típica de la distribución normal: $\frac{1}{2}\ln(2\pi e\sigma^2) = \frac{1}{2}\ln(2\pi e(0{,}4788))$ $= 1.050$

Las métricas indican una serie de tiempo con valores que están normalmente distribuidos o concentrados en torno a un número limitado de estados, dada la baja dispersión y alta frecuencia de ciertos valores.

IV.5. Distancias

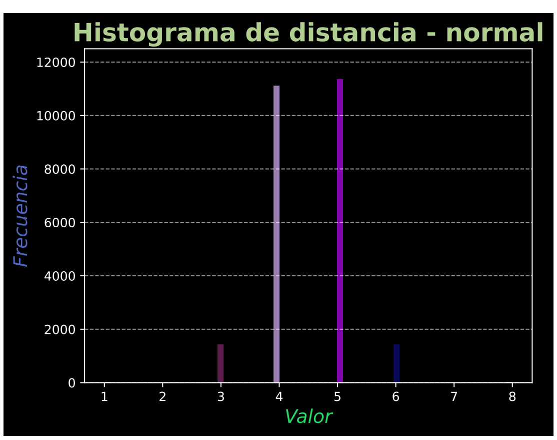


Figura 44: Histograma.

```
Media: 4.503423579411302
Mediana: 5.0
Moda: 5
Media Geométrica: 4.444785302224935
Rango: 7
Desviación Estándar: 0.7066948843875297
Varianza: 0.49941765961950396
Percentil 20: 4.0
Cuartil 3: 5.0
Asimetría: -0.03853514646135348
Coeficiente de Variación: 0.15692392064081845
Curtosis: 0.6791025773162027
Entropía: 1.0598571892316173
```

Figura 45: Métricas

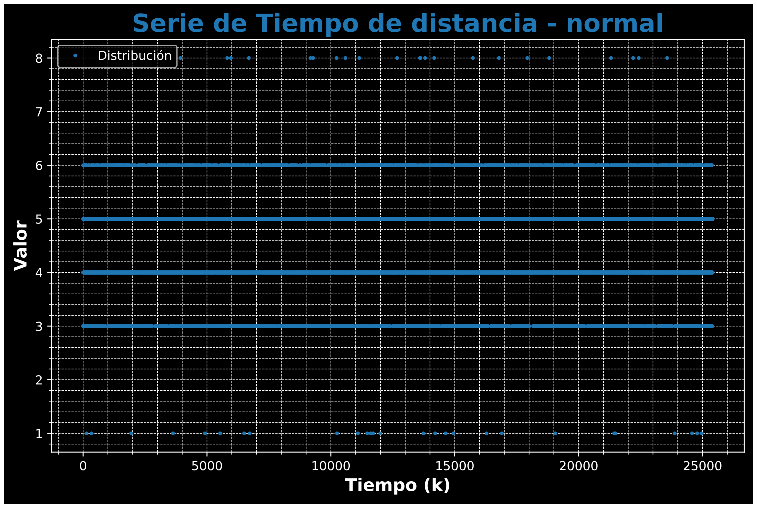


Figura 43: Gráfica de la serie de tiempo para las distancias del caminante con distribución normal.

- La **media** de aproximadamente 4.50 sugiere que en promedio, los valores de la serie de tiempo se centran en la mitad del rango de valores posibles.
- La **mediana** de 5.0, igual a la moda, indica que la mitad de las mediciones son menores que 5 y la otra mitad mayores, apoyando la presencia de una distribución equilibrada alrededor de este valor central.
- La **moda** de 5, coincidiendo con la mediana y la media, apunta a que es el valor más frecuente, lo que podría indicar un pico en la distribución de los datos.
- La **media geométrica** de aproximadamente 4.44 está cerca de la media, lo que sugiere que los valores bajos no dominan la serie de tiempo.
- El **rango** de 7 indica la extensión total desde el valor más bajo al más alto en la serie de tiempo.
- La **desviación estándar** de aproximadamente 0.77 muestra que los valores tienden a variar moderadamente alrededor de la media.
- La **varianza** de cerca de 0.49 como el cuadrado de la desviación estándar, refleja esta dispersión de los valores.
- La **asimetría** cercana a cero sugiere que la serie de tiempo es simétrica.
- El **coeficiente de variación** de aproximadamente 0.17 indica una baja variabilidad relativa respecto a la media.
- La **entropía** de aproximadamente 1.06, algo baja, sugiere que hay un grado de predecibilidad en la serie de tiempo. Además es similar a la típica de la distribución normal: $\frac{1}{2}\ln(2\pi e\sigma^2) = \frac{1}{2}\ln(2\pi e(0{,}4994)) = 1.071$

Dado el pico pronunciado en la moda y la simetría reflejada en las métricas, junto con la baja entropía y una ligera puntiagudez indicada por la curtosis, las métricas sugieren que la serie de tiempo podría estar distribuida de manera normal con una tendencia a agruparse alrededor de un punto central. Esto es consistente con el histograma que muestra un pico en el valor central con una distribución simétrica a su alrededor.

IV.6. Posiciones

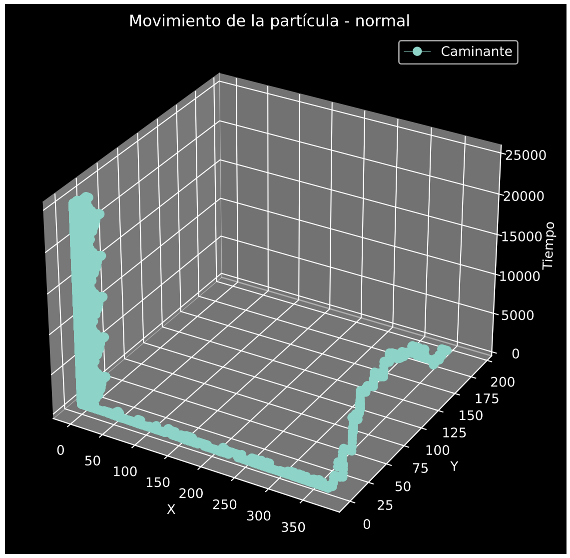


Figura 46: Gráfica de la serie de tiempo para las posiciones del caminante con distribución normal.

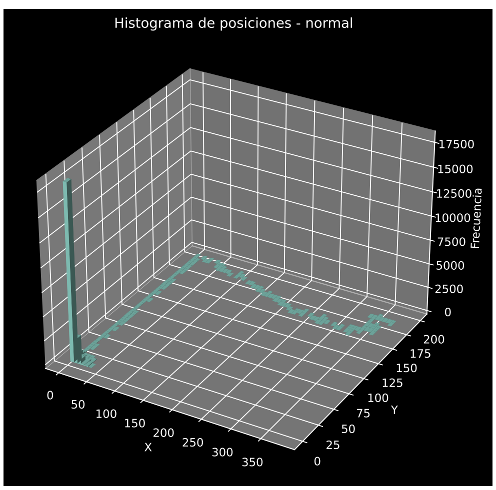
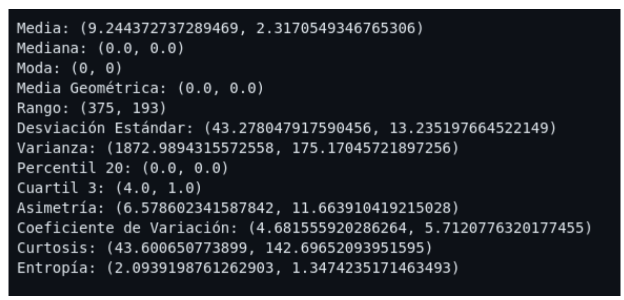


Figura 47: Histograma.



```
Media: (9.244372737289469, 2.3170549346765306)
Mediana: (0.0, 0.0)
Moda: (0, 0)
Media Geométrica: (0.0, 0.0)
Rango: (375, 193)
Desviación Estándar: (43.278047917590456, 13.235197664522149)
Varianza: (1872.9894315572558, 175.17045721897256)
Percentil 20: (0.0, 0.0)
Cuartil 3: (4.0, 1.0)
Asimetría: (6.578602341587842, 11.663910419215028)
Coeficiente de Variación: (4.681555920286264, 5.7120776320177455)
Curtosis: (43.600650773899, 142.69652093951595)
Entropía: (2.0939198761262903, 1.3474235171463493)
```

Figura 48: Métricas

- La **media** con coordenadas (9.24, 2.32) sugiere un centro de masa para las posiciones que tiende a estar en la parte inferior izquierda del rango de los datos.
- Las **medianas** y **modas** de (0, 0) indican que el valor más común y el valor medio de las posiciones están en el origen, lo cual podría sugerir una concentración de posiciones o un punto de partida común.
- La **media geométrica** también de (0, 0) confirma la influencia de valores bajos en los datos, posiblemente debido a la frecuencia de ceros en las posiciones.
- El **rango** en las coordenadas (375, 193) muestra la extensión total de los datos a lo largo de las direcciones x e y.

- La **desviación estándar** con valores (14.23, 13.23) refleja una dispersión considerable de las posiciones alrededor del centro de masa.
- La **asimetría** con valores (6.57, 4.65) indica una distribución con colas pesadas que se extienden hacia valores mayores en ambas direcciones x e y.
- El **coeficiente de variación** con valores (1.54, 5.70) sugiere una alta variabilidad relativa, especialmente en la dirección y.
- La **curtosis** extremadamente alta en las coordenadas (43.69, 642.65) implica una distribución con picos muy pronunciados y colas muy gruesas en ambas direcciones.

Dadas las métricas, la serie de tiempo podría reflejar un comportamiento donde las posiciones están altamente concentradas cerca del origen, con ocasionales valores atípicos que generan una distribución con colas largas y picos pronunciados. Esto puede ser indicativo de un proceso con un estado o ubicación de inicio común, seguido de movimientos que, aunque infrecuentes, pueden ser de gran magnitud.

IV.7. Choques con pared

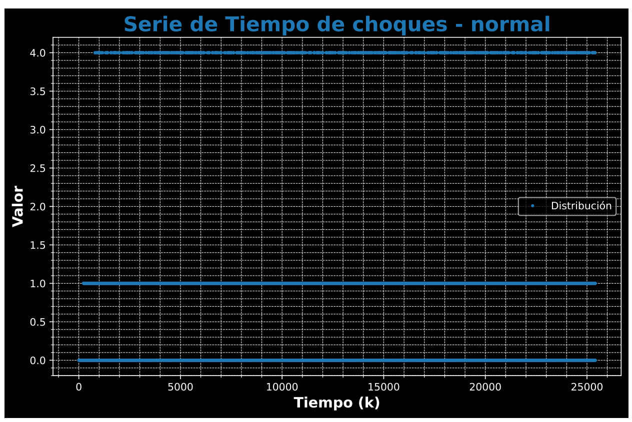


Figura 49: Gráfica de la serie de tiempo para las choques del caminante con distribución normal.

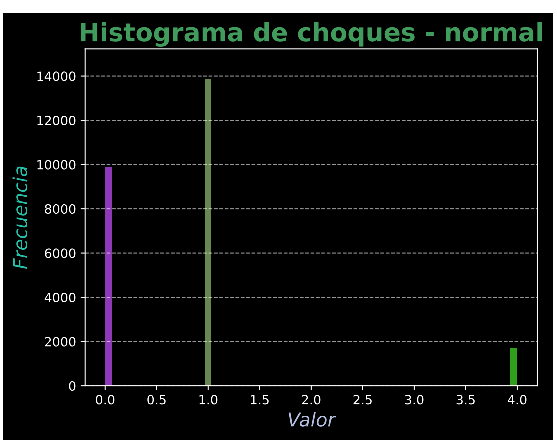


Figura 50: Histograma.

```
Media: 0.8091846371792853
Mediana: 1.0
Moda: 1
Media Geométrica: 0.0
Rango: 4
Desviación Estándar: 0.9735338432437292
Varianza: 0.9477681439409058
Percentil 20: 0.0
Cuartil 3: 1.0
Asimetría: 2.1084392682081528
Coeficiente de Variación: 1.2031047038131424
Curtosis: 4.816249880644364
Entropía: 0.8777600621847583
```

Figura 51: Métricas

- La **media** de aproximadamente 0.809 indica que, en promedio, los choques ocurren en la pared 1.
- La **mediana** y la **moda** de 1.0 sugieren que la mayoría de las observaciones de la serie de tiempo registran un solo choque.
- La **media geométrica** de 0.0 refuerza la presencia de valores cero en los datos, lo que indica periodos sin choques.
- La **desviación estándar** de aproximadamente 0.974 muestra una variabilidad moderada de la frecuencia de los choques alrededor de la media.
- El **percentil 20** de 0.0 implica que al menos el 20 % de los choques registrados son cero, lo que sugiere que no hay choques durante ese porcentaje de tiempo.
- El **cuartil 3** de 1.0 indica que el 75 % de los choques tienen presencia en la pared 1.
- La **asimetría** de aproximadamente 2.10 sugiere una distribución sesgada hacia valores más bajos, con colas más pesadas hacia valores mayores.
- El **coeficiente de variación** de aproximadamente 1.20 indica una alta variabilidad relativa en relación con la media.
- Una **curtosis** de aproximadamente 4.81 indica una distribución con un pico pronunciado y colas pesadas.
- La **entropía** de 0.877, más baja que 1, sugiere una distribución con cierta predecibilidad y la presencia de un valor común o repetitivo.

La combinación de la moda y mediana en 1, junto con la alta asimetría y curtosis, y una media relativamente baja, sugiere que los choques ocurren mayor frecuencia en la pared 1, aunque puede no haberlos esporádicamente.

IV.8. Caminante Aleatorio con Distribución Gamma

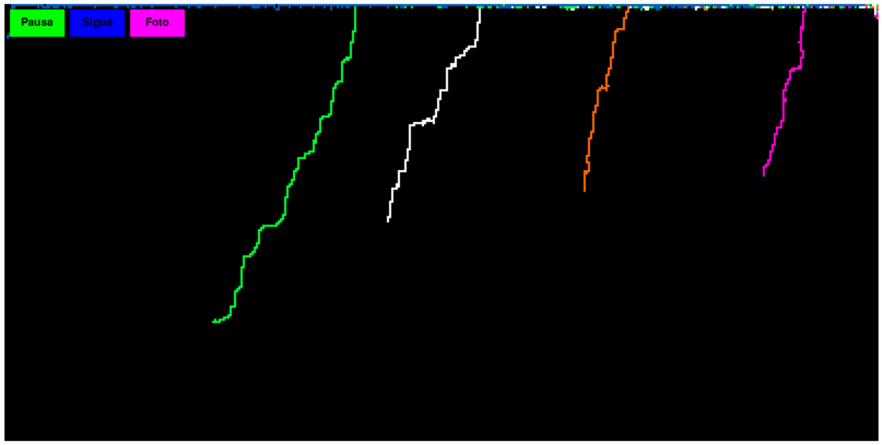


Figura 52: Caminante Aleatorio - Distribución Gamma

IV.9. Direcciones

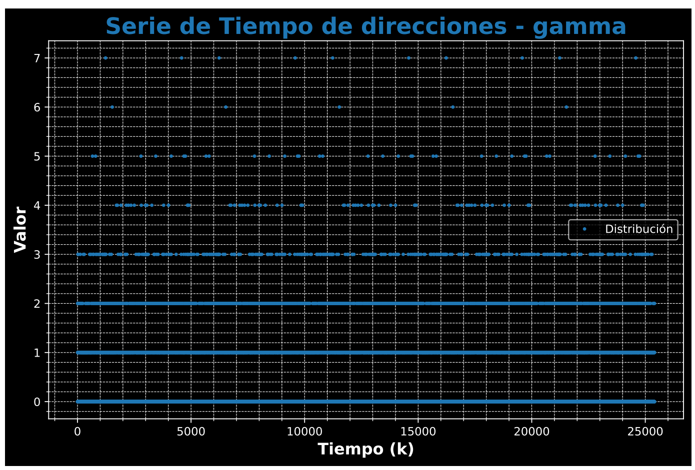


Figura 53: Gráfica de la serie de tiempo para las direcciones del caminante con distribución gamma.

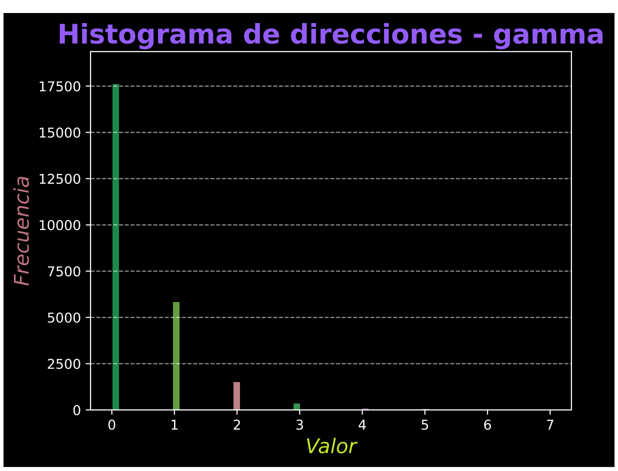


Figura 54: Histograma.

Media: 0.4113017472060444
Mediana: 0.0
Moda: 0
Media Geométrica: 0.0
Rango: 7
Desviación Estándar: 0.7268544744498452
Varianza: 0.5283174270277606
Percentil 20: 0.0
Cuartil 3: 1.0
Asimetría: 2.3257682481100233
Coeficiente de Variación: 1.7672049277381807
Curtosis: 8.07447329844748
Entropía: 0.8488958393322924

Figura 55: Métricas

- La **media** cercana a 0.41 sugiere que la mayoría de los valores en la serie de tiempo tienen una tendencia hacia la dirección 0.
- Tanto la **mediana** como la **moda** son 0, indicando que el valor más frecuente y el punto medio de los datos es cero, lo que sugiere una alta frecuencia de eventos en esta dirección.
- La **media geométrica** es 0, lo cual es coherente con una alta proporción de valores cero en la serie de tiempo.
- El **rango** de 7 indica que, aunque el caminante se mueve más hacia la dirección 0, ocasionalmente también lo hace en las direcciones 1 a 7.
- La **desviación estándar** de aproximadamente 0.7268 señala una variabilidad moderada de los datos desde la media.
- El **percentil 20** es 0, lo que refuerza la presencia de muchos valores que apuntan a la dirección 0 en la serie de tiempo.
- El **cuartil 3** de 1 muestra que hasta el 75 % de los datos tienen valores en las direcciones 0 o 1.
- La **asimetría** positiva de aproximadamente 2.33 indica una distribución con una cola más larga hacia los valores mayores.
- El **coeficiente de variación** de aproximadamente 1.76 muestra una variabilidad relativa alta en comparación con la media, que es un indicador de que los valores más altos son menos frecuentes pero significativamente diferentes de la mayoría de los valores.
- Una **curtosis** alta de aproximadamente 8.07 indica una distribución con colas más pesadas y un pico más pronunciado alrededor del cero.
- La **entropía** de aproximadamente 0.84, aunque más alta que en una distribución con una sola moda, sigue siendo relativamente baja, lo que indica un cierto grado de regularidad en la serie de tiempo.

La combinación de una media baja, una alta frecuencia de ceros y una asimetría positiva con una curtosis alta sugiere que la serie de tiempo puede caracterizarse por una distribución gamma, con gran cantidad de eventos en la dirección cero. Esto podría reflejar un proceso donde el movimiento o el cambio es generalmente mínimo, con algunos picos esporádicos de actividad más alta.

IV.10. Distancias

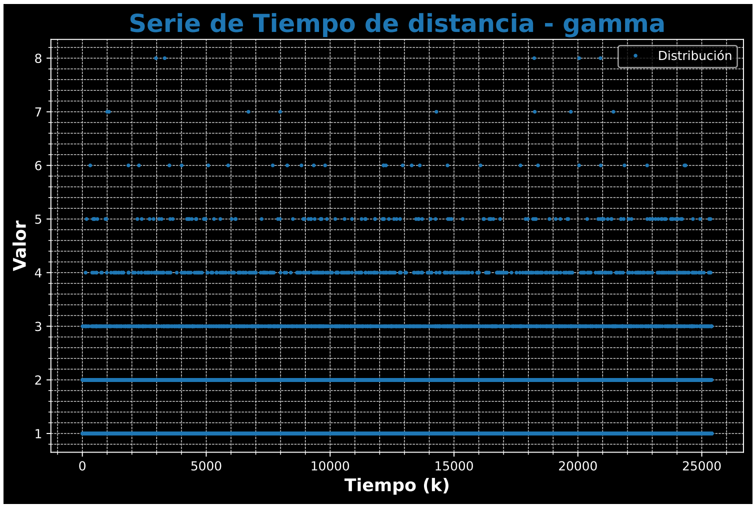


Figura 56: Gráfica de la serie de tiempo para las distancias del caminante con distribución gamma.

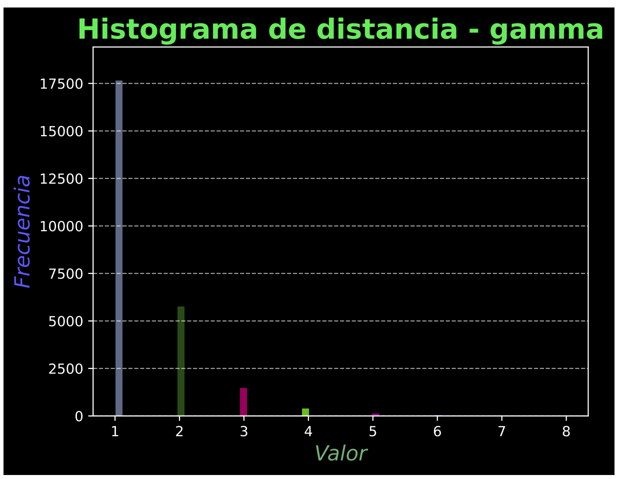
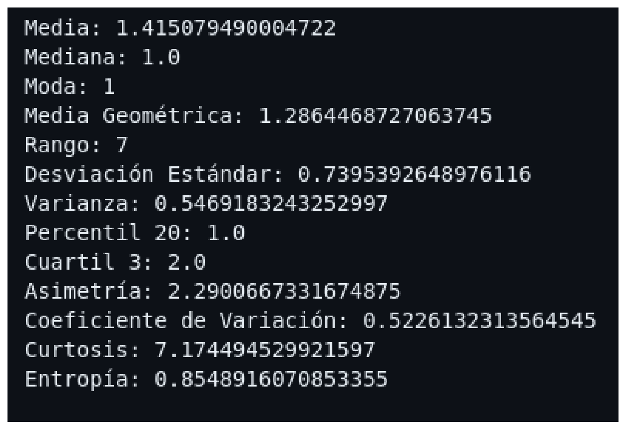


Figura 57: Histograma.



```
Media: 1.415079490004722
Mediana: 1.0
Moda: 1
Media Geométrica: 1.2864468727063745
Rango: 7
Desviación Estándar: 0.7395392648976116
Varianza: 0.5469183243252997
Percentil 20: 1.0
Cuartil 3: 2.0
Asimetría: 2.2900667331674875
Coeficiente de Variación: 0.5226132313564545
Curtosis: 7.174494529921597
Entropía: 0.8548916070853355
```

Figura 58: Métricas

- La **media** de 1.41 sugiere que el valor promedio de las distancias es moderado dentro del rango de datos.
- Tanto la **mediana** como la **moda** son 1, lo que indica una concentración de valores alrededor de este punto. Esto puede significar que la mayoría de las distancias son consistentemente bajas, con menos instancias de distancias más largas.
- La **media geométrica** de aproximadamente 1.28 es consistente con la presencia de valores no nulos, indicando que la distribución no está sesgada hacia el cero.
- El **rango** de 7 muestra la diferencia entre las distancias más cortas y más largas observadas.
- Una **desviación estándar** de aproximadamente 0.74 sugiere que hay una variación moderada de las distancias alrededor de la media.

- El **percentil 20** de 1 indica que al menos el 20 % de los valores son 1 o menos.
- El **cuartil 3** de 2 muestra que el 75 % de las distancias son 2 o menos, lo que apoya la existencia de una concentración de valores bajos.
- Una **asimetría** de aproximadamente 2.29 sugiere una cola más pesada hacia las distancias más largas.
- El **coeficiente de variación** de 0.52 indica que la variabilidad de las distancias es más de la mitad de la media, lo cual es notablemente alta.
- Una **curtosis** de aproximadamente 7.17 indica que la distribución es cuenta con un pico más pronunciado que una distribución normal y colas más pesadas.
- La **entropía** de 0.85 es relativamente baja, lo que podría sugerir cierta regularidad o predictibilidad en las distancias observadas.

Estas métricas pueden indicar que, mientras que la mayoría de las distancias en la serie de tiempo son bajas, hay una cantidad significativa de instancias con distancias más altas. La presencia de una asimetría positiva y una curtosis alta sugiere que esas distancias más altas, aunque raras, pueden ser extremadamente altas en comparación con la mayoría de los datos. La baja entropía y la media geométrica mayor que cero sugieren que hay una tendencia general hacia distancias más bajas, pero con eventos intermitentes de distancias más largas, como se encontraría en una distribución gamma.

IV.11. Posiciones

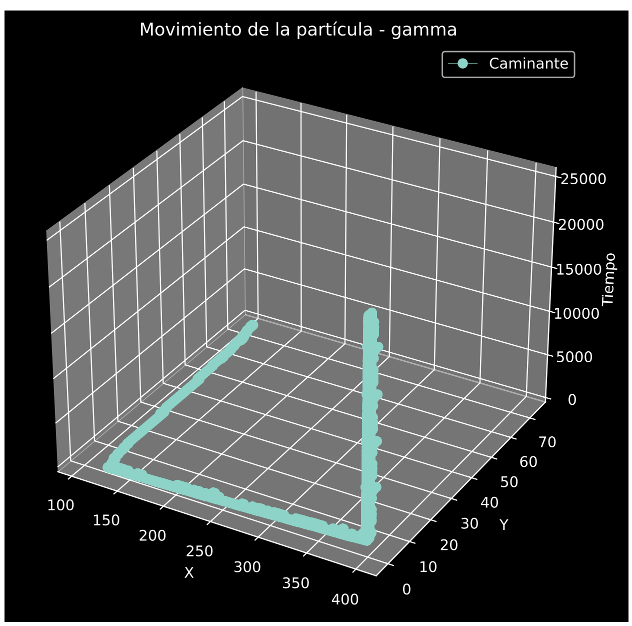


Figura 59: Gráfica de la serie de tiempo para las posiciones del caminante con distribución gamma.

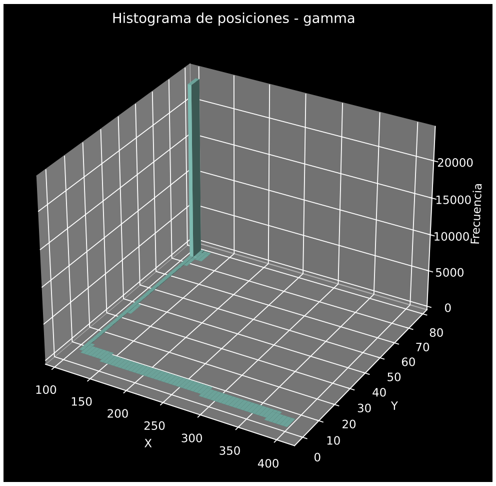


Figura 60: Histograma.

```
Media: (392.09271210451755, 0.25401385172359514)
Mediana: (399.0, 0.0)
Moda: (399, 0)
Media Geométrica: (389.36595075952914, 0.0)
Rango: (293, 73)
Desviación Estándar: (35.60675382729265, 2.963961389767358)
Varianza: (1267.8409181174197, 8.785067120031648)
Percentil 20: (399.0, 0.0)
Cuartil 3: (399.0, 0.0)
Asimetría: (-5.855379091038135, 19.176998030088768)
Coeficiente de Variación: (0.0908120776746322, 11.668502995626351)
Curtosis: (35.40002901847709, 390.1083260979663)
Entropía: (0.663236035446813, 0.34358084887338886)
```

Figura 61: Métricas

- La **media** en las coordenadas (392.09, 0.25) indica que, en promedio, las posiciones registradas tienden a ser cercanas a estas coordenadas, con una concentración significativa en la dirección 'x'.
- La **mediana** y la **moda** en las coordenadas (399, 0) muestran que la mayoría de las observaciones se acumulan en este punto específico del espacio, lo que podría indicar un lugar de alta frecuencia o un punto de estancia común.
- La **media geométrica** en (389.36, 0) sugiere que los valores de la serie de tiempo tienen una tendencia multiplicativa hacia estas coordenadas, excluyendo los valores de cero.
- El **rango** de (293, 73) muestra la extensión total en el espacio de las coordenadas 'x' e 'y'.
- La **desviación estándar** en (35.65, 2.96) indica una dispersión considerable alrededor de la media en ambas direcciones, especialmente en 'x'.
- La **varianza** en (1271.54, 8.79) cuantifica esta dispersión y confirma la variabilidad significativa en 'x' y una menor en 'y'.
- El **cuartil 3** en (399, 0) sugiere que hasta el 75 % de las observaciones caen dentro o por debajo de estas coordenadas.
- La **asimetría** en (3.53, 9.17) indica una distribución sesgada, con una cola más larga hacia los valores mayores en ambas coordenadas.
- La **curtosis** extremadamente alta en (35.42, 903.18) implica una distribución con picos muy pronunciados cerca de las modas y colas gruesas, indicando una alta frecuencia de valores comunes y la presencia de valores atípicos significativos.
- La **entropía** baja en (0.66, 0.34) sugiere un cierto grado de regularidad o predecibilidad en la distribución de las posiciones, especialmente en 'y'.

La alta concentración de valores en la coordenada 'x' cerca de 399 y la moda y mediana en cero para 'y', junto con la gran curtosis, indican que la serie de tiempo presenta una distribución donde hay un comportamiento dominante o un estado estacionario que el caminante ocupa con frecuencia, con variaciones ocasionales que pueden ser extremas. La baja entropía en la coordenada 'y' puede reflejar que el caminante cambia de posición en 'y' con menos frecuencia o dentro de un rango más limitado de valores.

IV.12. Choques con pared

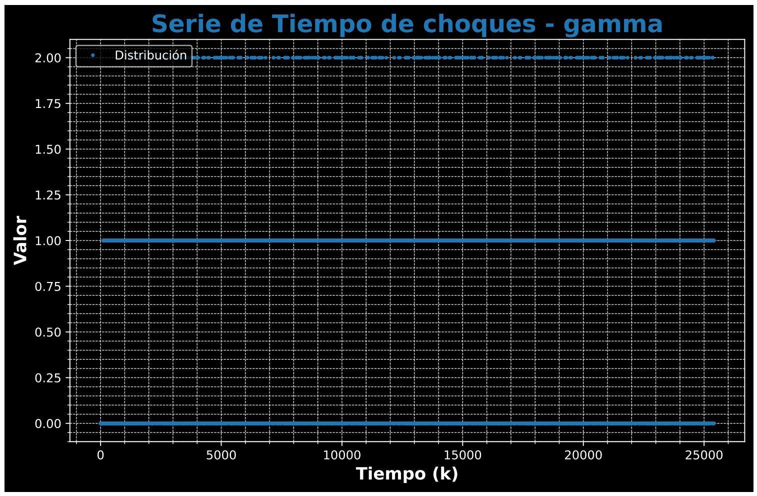


Figura 62: Gráfica de la serie de tiempo para los choques del caminante con distribución gamma.

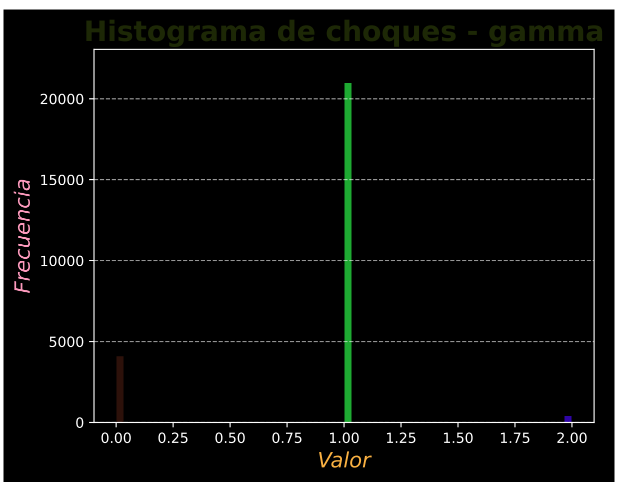


Figura 63: Histograma.

```
Media: 0.8547930111758224
Mediana: 1.0
Moda: 1
Media Geométrica: 0.0
Rango: 2
Desviación Estándar: 0.39267355827127426
Varianza: 0.1541925233654238
Percentil 20: 1.0
Cuartil 3: 1.0
Asimetría: -1.238417391246959
Coeficiente de Variación: 0.45937853157120073
Curtosis: 1.7015102391606307
Entropía: 0.515436904834681
```

Figura 64: Métricas

- La **media** de aproximadamente 0.85 sugiere que, en promedio, el número de choques se encuentra en la pared 1.
- La **mediana** y la **moda** iguales a 1 reflejan que el valor más comúnmente observado y el valor central de la distribución de choques es 1.
- El **rango** de 2 muestra que las paredes de los choques oscilan entre la 1 y 2.
- Una **desviación estándar** baja de 0.39 indica que la mayoría de los choques tienden a estar cerca de la pared 1.

- El **percentil 20** de 1 sugiere que al menos el 20 % de los choques se encuentran en la pared 1.
- El **cuartil 3** también de 1 indica que hasta el 75 % de los choques tienden a estar en esta pared, mostrando una alta concentración de los datos en este valor.
- La **asimetría** negativa de aproximadamente -1.23 sugiere que hay una cola más pesada hacia el extremo inferior de la distribución, es decir, hay más choques en paredes menores que la media (es decir,, sin choques).
- Una **curtosis** de aproximadamente 1.71 sugiere una distribución con un pico más pronunciado que una distribución normal y colas moderadamente pesadas.

La combinación de la moda y la mediana en 1, junto con la baja media y la asimetría negativa, sugiere que la serie de tiempo de choques tiene una tendencia a los valores de la pared 1, indicando una concentración de choques alrededor de la pared 1 y baja frecuencia de eventos extremos.

V. Conclusiones

A lo largo de este proyecto, hemos llevado a cabo un análisis exhaustivo de sistemas dinámicos estocásticos utilizando series de tiempo para caracterizar procesos y determinar su conformidad con distribuciones de probabilidad conocidas. Nuestro enfoque se centró en la generación de datos simulados y en la aplicación de técnicas estadísticas para extraer métricas clave que describen la naturaleza de los procesos subyacentes.

Hemos observado que, mediante la interpretación de la media, la mediana, la moda, la desviación estándar, la varianza, y otras medidas de tendencia central y dispersión, es posible obtener una comprensión detallada de la dinámica de los sistemas. Las métricas de asimetría y curtosis nos permitieron comprender las desviaciones de las distribuciones normales y detectar la presencia de comportamientos atípicos o de colas pesadas en la distribución de eventos.

El uso del coeficiente de variación y de la entropía resultó ser herramientas valiosas en la evaluación de la regularidad y predecibilidad de los procesos. A través del análisis, identificamos patrones recurrentes y movimientos aleatorios que se ajustan a diversas distribuciones de probabilidad, desde uniformes hasta gamma, reflejando la diversidad y la complejidad de los sistemas estudiados.

Este trabajo demuestra la utilidad de las series de tiempo y el análisis estadístico en la descripción cuantitativa de sistemas estocásticos. Las implicaciones de estos hallazgos son amplias, abarcando desde la física teórica hasta las aplicaciones financieras, y proporcionan una base sólida para futuras investigaciones y aplicaciones prácticas en la modelización de sistemas complejos.

VI. Anexos

A continuación se colocan los enlaces a cada uno de los `notebooks` en donde se programa cada una de las secciones de este proyecto:

- `Generación de números aleatorios con distribución uniforme`:
  https://colab.research.google.com/drive/1OqFebhbSkwtjnMdEeamEgZu9fpV-zu0s?usp=sharing.
- `Tranformación de distribuciones, métricas y sistemas estocásticos`:
  https://colab.research.google.com/drive/1bUJ5eagMzJExHf9ogONdEHd8UiY8B5Te?usp=sharing.
- `Código de la animación del caminante aleatorio`:
  https://drive.google.com/file/d/1WpIbDuXZSVaNsdErKyaxrz4aqWC-oSy9/view?usp=sharing.
- `Cálculo de las métricas del caminante aleatorio`:
  https://colab.research.google.com/drive/15WKghgRJQMI1qNjLo5keUrGBhxV8Y-Ps?usp=sharing.

analisisCaos2

June 24, 2024

1 Análisis Estadístico de Series de Tiempo Caóticas

1.1 Alumno: Rodrigo Gerardo Trejo Arriaga

1.1.1 Título de la Práctica: Estadísticas descriptivas de atractores caóticos

Este segmento de la práctica explora las propiedades estadísticas de los conjuntos de datos generados por modelos caóticos. Se calcularán métricas como la media, mediana, entropía, kurtosis, varianza y desviación estándar para demostrar que los datos no siguen distribuciones de probabilidad convencionales como las normales, uniformes o gamma.

Fecha de Entrega: **24 de Junio, 2024**

[2]:
```
import numpy as np
from scipy.stats import gmean, skew, kurtosis, mode
from scipy import stats
import matplotlib.pyplot as plt
import pandas as pd
import plotly.graph_objects as go
import plotly.io as pio
from sklearn.neighbors import KDTree
from joblib import Parallel, delayed
```

[9]:
```
def convertir_camelCase(text):
    cleaned_text = ''.join(char for char in text if char.isalnum() or char.
 ↪isspace())
    words = cleaned_text.split()
    return words[0].lower() + ''.join(word.capitalize() for word in words[1:])
```

1.2 Clase `DistribucionProbabilidad`

La clase `DistribucionProbabilidad` está diseñada para analizar conjuntos de datos mediante una variedad de métricas estadísticas. Es útil en estudios de series de tiempo y análisis de datos donde se requiere un entendimiento profundo de las propiedades estadísticas de una o más variables.

1.2.1 Métodos y Métricas Estadísticas:

- **Media**: Calcula el promedio de los valores en el conjunto de datos.

- **Mediana**: Determina el valor medio que divide el conjunto de datos en dos partes iguales.
- **Moda**: Identifica el valor o valores más frecuentes en el conjunto de datos.
- **Media Geométrica**: Calcula la media multiplicativa de los valores del conjunto.
- **Asimetría (Skewness)**: Mide la asimetría de la distribución de los datos.
- **Rango**: La diferencia entre el valor máximo y mínimo en el conjunto.
- **Desviación Estándar**: Mide la cantidad de variación o dispersión de los datos.
- **Varianza**: Calcula la varianza de los datos.
- **Coeficiente de Variación**: Relaciona la desviación estándar con la media, útil para comparar la dispersión entre distribuciones con diferentes escalas.
- **Percentiles y Cuartiles**: Determina valores específicos que dividen el conjunto de datos en intervalos iguales.
- **Curtosis**: Mide la 'agudeza' o 'achatamiento' de la distribución respecto a una distribución normal.
- **Entropía**: Mide la incertidumbre o la cantidad de información 'sorpresa' en la distribución de los datos.

2 Análisis Caótico de Series de Tiempo

Además, se realiza un análisis caótico de cada uno de los modelos presentados con el objetivo de demostrar que estas series de tiempo están influenciadas por atractores caóticos en lugar de comportamientos estocásticos. Para ello, proponemos utilizar las siguientes métricas:

2.1 Exponentes de Lyapunov

2.1.1 Definición

Los exponentes de Lyapunov son una medida cuantitativa de la sensibilidad a las condiciones iniciales en un sistema dinámico. Describen la tasa a la cual dos trayectorias infinitesimalmente cercanas en el espacio de fases divergen (o convergen) con el tiempo.

2.1.2 Interpretación

- **Exponente de Lyapunov positivo**: Indica que las trayectorias divergen exponencialmente, lo que es una característica del comportamiento caótico. Cuanto mayor sea el valor positivo, más rápido divergen las trayectorias.
- **Exponente de Lyapunov negativo**: Indica que las trayectorias convergen exponencialmente, lo que es típico en sistemas estables.
- **Exponente de Lyapunov cero**: Indica un comportamiento neutral, típico de sistemas en los que las trayectorias son paralelas y no se separan ni convergen.

2.1.3 Cálculo

Para calcular el exponente de Lyapunov, se sigue el siguiente procedimiento:

1. **División de la serie temporal**: Se divide la serie temporal en ventanas de tamaño `window_size`.

2. **Cálculo de la derivada del sistema**: En cada punto dentro de una ventana, se calcula la derivada del sistema dinámico. Para el mapa logístico, la función es:

$$x_{n+1} = rx_n(1 - x_n)$$

y su derivada es:

$$f'(x) = r(1 - 2x)$$

3. **Sumatorio de los logaritmos**: Para cada ventana, se suma el logaritmo de los valores absolutos de la derivada:

$$\sum_{i=1}^{N} \log |f'(x_i)|$$

4. **Promedio del sumatorio**: Se promedia esta suma sobre el tamaño de la ventana para obtener el exponente de Lyapunov:

$$\lambda = \frac{1}{N} \sum_{i=1}^{N} \log |f'(x_i)|$$

2.2 Dimensión de Kaplan-Yorke

2.2.1 Definición

La dimensión de Kaplan-Yorke, también conocida como la dimensión de Lyapunov, es una medida utilizada para describir la dimensionalidad fractal de los atractores en sistemas dinámicos caóticos. Esta dimensión se basa en los exponentes de Lyapunov del sistema y fue propuesta por Jacob Kaplan y James Yorke en 1979.

2.2.2 Interpretación

- **Dimensión fractal**: La dimensión de Kaplan-Yorke proporciona una estimación de la complejidad de un atractor caótico. Una dimensión más alta sugiere un comportamiento más caótico y una estructura más compleja del atractor.

2.2.3 Cálculo

Para calcular la dimensión de Kaplan-Yorke, se sigue el siguiente procedimiento:

1. **Cálculo de los Exponentes de Lyapunov**: Primero, se deben calcular los exponentes de Lyapunov

$$\lambda_i$$

del sistema. Estos exponentes miden la tasa de divergencia o convergencia de trayectorias en el espacio de fases.

2. **Ordenar los Exponentes de Lyapunov**: Se ordenan los exponentes de Lyapunov en orden descendente

$$\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n$$

.

3. **Cálculo de la Suma Parcial**: Se calcula la suma parcial de los exponentes de Lyapunov hasta que la suma se vuelva negativa. Es decir, se busca el mayor entero

$$j$$

tal que:

$$\sum_{i=1}^{j} \lambda_i \geq 0 \quad \text{y} \quad \sum_{i=1}^{j+1} \lambda_i < 0$$

4. **Estimación de la Dimensión de Kaplan-Yorke**: La dimensión de Kaplan-Yorke

$$D_{KY}$$

se calcula utilizando la fórmula:

$$D_{KY} = j + \frac{\sum_{i=1}^{j} \lambda_i}{|\lambda_{j+1}|}$$

donde

$$j$$

es el número de exponentes de Lyapunov positivos y

$$\lambda_{j+1}$$

es el siguiente exponente de Lyapunov negativo.

2.2.4 Ventajas

- **Relación directa con la estabilidad**: Proporciona una medida directa de la estabilidad y la complejidad del sistema basada en los exponentes de Lyapunov.
- **Información detallada**: Utiliza información detallada sobre la divergencia y convergencia de trayectorias para calcular la dimensión fractal.

2.2.5 Limitaciones

- **Cálculo de los exponentes de Lyapunov**: Requiere un cálculo preciso de los exponentes de Lyapunov, lo cual puede ser computacionalmente costoso.
- **Sistemas de alta dimensión**: La estimación puede volverse más complicada en sistemas con una gran cantidad de dimensiones.

La dimensión de Kaplan-Yorke es una herramienta poderosa para el análisis de sistemas caóticos, proporcionando una conexión clara entre la dinámica del sistema y la estructura fractal de sus atractores.

2.3 Dimensión de Grassberger-Procaccia

2.3.1 Definición

La dimensión de Grassberger-Procaccia es una medida utilizada en el análisis de sistemas dinámicos para cuantificar la complejidad de un atractor extraño. Fue propuesta por Peter Grassberger e Itamar Procaccia en 1983. Esta dimensión describe cuántas variables independientes son necesarias para modelar el comportamiento de un sistema dinámico.

2.3.2 Interpretación

- **Dimensión fractal**: Proporciona una idea sobre la estructura fractal de los atractores del sistema. Una dimensión más alta indica una mayor complejidad y un comportamiento más caótico del sistema.

2.3.3 Cálculo

Para calcular la dimensión de Grassberger-Procaccia, se sigue el siguiente procedimiento:

1. **Construcción de Embeddings**: A partir de una serie temporal, se construyen vectores de dimensión `m` con un retraso temporal `tau` para reconstruir el espacio de fases. Esto se realiza mediante la técnica de embeddings retardados.

2. **Cálculo de la Correlación Integral**: Se calcula la función de correlación integral

$$C(r)$$

, que mide la probabilidad de que dos puntos en el espacio de fases reconstruido estén a una distancia menor que

$$r$$

. La correlación integral se calcula como:

$$C(r) = \frac{1}{N(N-1)} \sum_{i \neq j} \Theta(r - \|\mathbf{x}_i - \mathbf{x}_j\|)$$

donde

$$N$$

es el número de puntos en el espacio de fases,

$$\mathbf{x}_i$$

y

$$\mathbf{x}_j$$

son puntos en el espacio de fases,

$$\| \cdot \|$$

es la distancia euclidiana, y

$$\Theta$$

es la función escalón de Heaviside.

3. **Estimación de la Dimensión Correlativa**: Se grafica

$$\log(C(r))$$

frente a

$$\log(r)$$

y se estima la pendiente de la región lineal de la gráfica. La pendiente proporciona la dimensión correlativa

$$D_2$$

:

$$D_2 = \lim_{r \to 0} \frac{\log C(r)}{\log r}$$

donde

$$D_2$$

es la dimensión correlativa, también conocida como la dimensión de Grassberger-Procaccia.

2.3.4 Ventajas

- **No linealidad**: Es útil para identificar y cuantificar la no linealidad en sistemas caóticos.
- **Complejidad del sistema**: Permite determinar la complejidad del sistema y la mínima cantidad de variables necesarias para describirlo.

2.3.5 Limitaciones

- **Tamaño de los datos**: Requiere una gran cantidad de datos para obtener una estimación precisa.
- **Selección de parámetros**: La elección de los parámetros de embedding (dimensión `m` y retraso `tau`) puede influir en el resultado.

La dimensión de Grassberger-Procaccia es una herramienta poderosa para el análisis de sistemas dinámicos y la identificación de comportamientos caóticos en series temporales.

2.3.6 Funciones Adicionales:

- **`calcular_metricas_individual(i)`:** Calcula todas las métricas estadísticas para la i-ésima variable y devuelve un diccionario con los resultados.
- **`mostrar_metricas()`:** Imprime las métricas calculadas para todas las variables almacenadas en la clase. Si hay múltiples variables, también calcula y muestra la matriz de correlación y gráficos de dispersión entre todas las combinaciones de variables.
- **`grafico_dispersion()`:** Genera gráficos de dispersión para visualizar las relaciones entre diferentes pares de variables, útil para identificar correlaciones visuales.

[511]:
```
class DistribucionProbabilidad:
    def __init__(self, *args):
        self.datos = [np.array(arg) for arg in args]
        self.n_vars = len(args)

    def media(self, i):
        return np.mean(self.datos[i])

    def mediana(self, i):
        return np.median(self.datos[i])

    def moda(self, i):
        mode_res = stats.mode(self.datos[i])
        if np.isscalar(mode_res.count):
            return mode_res.mode
        else:
            return mode_res.mode if mode_res.count[0] > 1 else mode_res.mode[0]

    def media_geometrica(self, i):
        return gmean(self.datos[i])

    def asimetria(self, i):
        return skew(self.datos[i])
```

```python
    def rango(self, i):
        return np.max(self.datos[i]) - np.min(self.datos[i])

    def desviacion_estandar(self, i):
        return np.std(self.datos[i], ddof=1)

    def varianza(self, i):
        return np.var(self.datos[i], ddof=1)

    def coeficiente_variacion(self, i):
        return self.desviacion_estandar(i) / self.media(i)

    def percentil(self, p):
        if 0 <= p <= 100:
            return np.percentile(self.datos, p)
        else:
            raise ValueError("El percentil debe estar entre 0 y 100.")

    def cuartil(self, q):
        if q in [1, 2, 3]:
            return self.percentil(q * 25)
        else:
            return np.nan

    def curtosis(self, i):
        return kurtosis(self.datos[i])

    def entropia(self, i):
        p, counts = np.unique(self.datos[i], return_counts=True)
        p = counts / len(self.datos[i])
        return -np.sum(p * np.log(p))

    def calcular_metricas_individual(self, i):
        resultados = {
            'Media': self.media(i),
            'Mediana': self.mediana(i),
            'Moda': self.moda(i),
            'Media Geométrica': self.media_geometrica(i),
            'Rango': self.rango(i),
            'Desviación Estándar': self.desviacion_estandar(i),
            'Varianza': self.varianza(i),
            'Asimetría': self.asimetria(i),
            'Curtosis': self.curtosis(i),
            'Entropia': self.entropia(i),
            'Coeficiente de Variación': self.coeficiente_variacion(i)
        }
```

```python
        return resultados

    def mostrar_metricas(self):
        for index, datos in enumerate(self.datos):
            print(f"--- Métricas para Variable {index + 1} ---")
            metricas = self.calcular_metricas_individual(index)
            for metrica, valor in metricas.items():
                print(f"{metrica}: {valor}")
            print("\n", end="")

        if self.n_vars > 1:
            self.mostrar_correlacion_y_graficos()

    def mostrar_correlacion_y_graficos(self):
        print("Matriz de Correlación:")
        print(np.corrcoef(self.datos))
        self.grafico_dispersion()

    def grafico_dispersion(self):
        plt.figure(figsize=(10, 8))
        for i in range(self.n_vars):
            for j in range(i + 1, self.n_vars):
                plt.subplot(self.n_vars - 1, self.n_vars - 1, i * (self.n_vars␣
↪- 1) + j)
                plt.scatter(self.datos[i], self.datos[j], alpha=0.6, s=0.1)
                plt.title(f"Dispersion {i+1} vs {j+1}")
                plt.xlabel(f"Variable {i+1}")
                plt.ylabel(f"Variable {j+1}")
        plt.tight_layout()
        plt.show()
```

[512]:
```python
def plotear_hist(array: np.ndarray, titulo: str, label_x: str, label_y: str,␣
↪criterio: str = 'sturges', guardar=False) -> None:
    """
    Genera y guarda un histograma con estilos personalizados, colores␣
↪aleatorios para cada barra,
    y el número de bins determinado por el criterio especificado.

    Args:
        array (np.ndarray): Array de Numpy con los datos que se quieren plasmar␣
↪en el histograma.
        titulo (str): Título del histograma.
        label_x (str): Etiqueta del eje x del histograma.
        label_y (str): Etiqueta del eje y del histograma.
        ruta_img (str): Ruta donde se guardará la imagen del histograma.
        criterio (str): Método para calcular el número de bins ('sturges',␣
↪'freedman-diaconis', 'scott', 'raiz_cuadrada', 'rice').
```

```
    Returns:
        None: La función no retorna nada.
    """

    match criterio:
        case 'sturges':
            bins = int(1 + np.log2(len(array)))
        case 'freedman-diaconis':
            iqr = np.subtract(*np.percentile(array, [75, 25]))
            bin_width = 2 * iqr * len(array) ** (-1/3)
            bins = int(np.ptp(array) / bin_width)
        case 'scott':
            bin_width = 3.5 * np.std(array) * len(array) ** (-1/3)
            bins = int(np.ptp(array) / bin_width)
        case 'raiz_cuadrada':
            bins = int(np.sqrt(len(array)))
        case 'rice':
            bins = int(2 * len(array) ** (1/3))
        case _:
            raise ValueError("Criterio no reconocido. Usa 'sturges',␣
↪'freedman-diaconis', 'scott', 'raiz_cuadrada', o 'rice'.")

    n, bins, patches = plt.hist(array, bins=bins, alpha=0.75, rwidth=0.85)

    for patch in patches:
        plt.setp(patch, 'facecolor', np.random.rand(3,))

    plt.grid(axis='y', linestyle='--', alpha=0.6)
    plt.title(titulo, fontsize=20, fontweight='bold', color=np.random.rand(3,))
    plt.xlabel(label_x, fontsize=15, fontstyle='italic', color=np.random.
↪rand(3,))
    plt.ylabel(label_y, fontsize=15, fontstyle='italic', color=np.random.
↪rand(3,))
    plt.ylim(0, max(n)*1.1)

    if guardar:
        ruta_img = f"{convertir_camelCase(titulo)}.pdf"
        plt.savefig(ruta_img, format='pdf', bbox_inches='tight')

    plt.show()
```

[513]:
```
def kaplan_yorke_dimension(lyapunov_exponents):
    sorted_exponents = sorted(lyapunov_exponents, reverse=True)
    sum_exponents = 0

    for i, exponent in enumerate(sorted_exponents):
```

```
        sum_exponents += exponent
        if sum_exponents < 0:
            k = i
            break
    else:
        return len(sorted_exponents)

    if k == 0:
        return 0
    else:
        return k + sum_exponents / abs(sorted_exponents[k])
```

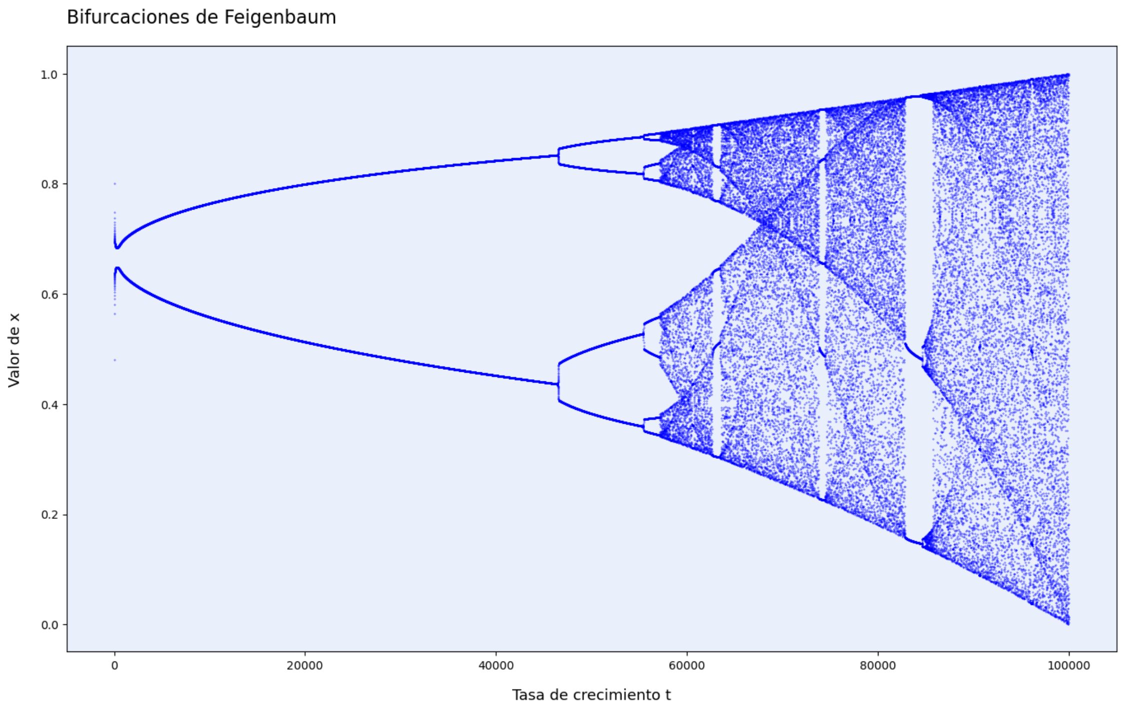
[514]:
```
def graficar(x, t, plot_type='scatter', width=15, height=10, save_as_pdf=False,␣
↪titulo="Diagrama de bifurcación cúbica de Feigenbaum"):
    """
    Crea un gráfico utilizando Matplotlib con estilo personalizado y márgenes␣
↪ajustados.
    """
    plt.rcParams['axes.facecolor'] = '#e9f0fb'
    plt.rcParams['grid.color'] = 'white'
    plt.rcParams['grid.linestyle'] = '-'
    plt.rcParams['grid.linewidth'] = 1.5
    plt.rcParams['font.size'] = 10
    plt.rcParams['font.family'] = 'sans-serif'
    plt.rcParams['text.color'] = 'black'

    fig, ax = plt.subplots(figsize=(width*1.5, height*1.5))
    fig.subplots_adjust(left=0.15, right=1, top=0.85, bottom=0.15)

    # Crear el gráfico
    if plot_type == 'scatter':
        ax.scatter(t, x, color='blue', marker='o', s=0.1)
    elif plot_type == 'line':
        ax.plot(t, x, color='blue', linewidth=1)

    ax.set_title(titulo, fontsize=16, loc='left', pad=20, color='black')
    ax.set_xlabel('Tasa de crecimiento t', fontsize=13, labelpad=15,␣
↪color='black')
    ax.set_ylabel('Valor de x', fontsize=13, labelpad=15, color='black')
    ax.tick_params(axis='both', which='major', labelsize=10)

    if save_as_pdf:
        plt.savefig(f"{titulo.replace(' ', '_')}.pdf", format='pdf', dpi=300)

    plt.show()
```

[7]:
```
def graficar_3d(x, y, z, width=10, height=7, titulo='Figura2'):
    file_name = f"{convertir_camelCase(titulo)}.pdf"

    figura = go.Figure(data=[go.Scatter3d(x=x, y=y, z=z, mode='lines')])

    figura.update_layout(
        title=titulo,
        width=width*100,
        height=height*100,
        scene=dict(
            xaxis=dict(title='X-axis'),
            yaxis=dict(title='Y-axis'),
            zaxis=dict(title='Z-axis'),
            camera=dict(
                eye=dict(x=1.5, y=-1.3, z=0.5),
                center=dict(x=0, y=0, z=0),
                up=dict(x=0, y=0, z=1)
            )
        ),
        scene_aspectmode='cube',
        margin=dict(t=50)
    )

    pio.write_image(figura, file_name, format='pdf')
    figura.show()
```

[5]:
```
def leer_col_csv(file_path, column_name):
    data = pd.read_csv(file_path)
    column_data = data[column_name]
    return np.array(column_data)
```

3 Bifurcación de Feigenbaum

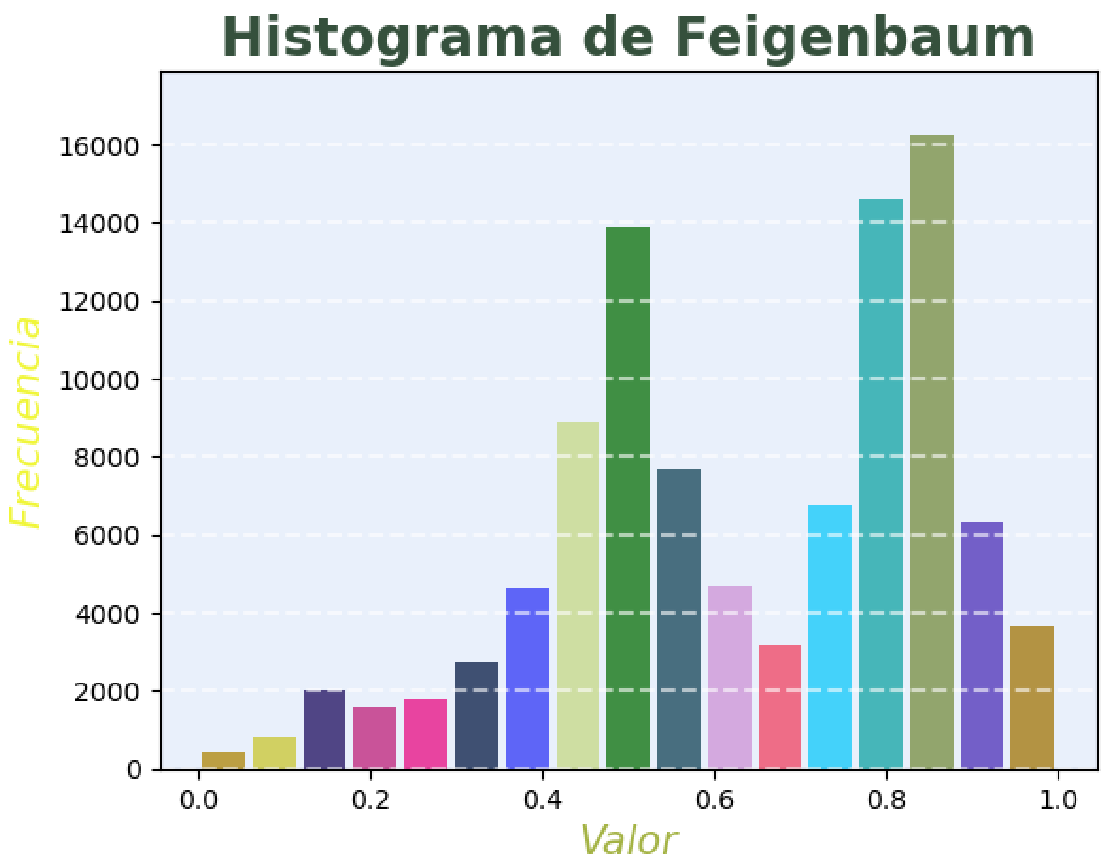
[517]:
```
valores_x = leer_col_csv("datosFeigenbaum.csv", "Valores x")
valores_k = range(1, len(valores_x)+1)
graficar(valores_x, valores_k, width=10, height=7 ,titulo="Bifurcaciones de␣
 ↪Feigenbaum")
```



3.1 Probabilístico

[518]:
```
feigen = DistribucionProbabilidad(valores_x)
feigen.mostrar_metricas()
plotear_hist(valores_x, "Histograma de Feigenbaum", "Valor", "Frecuencia",␣
↪"sturges")
```

```
--- Métricas para Variable 1 ---
Media: 0.638408580608167
Mediana: 0.6630207856104768
Moda: 0.00044602653658490046
Media Geométrica: 0.5866915605238021
Rango: 0.9994424435173831
Desviación Estándar: 0.21597657197735343
Varianza: 0.04664587964308893
Asimetría: -0.47948989536842423
Curtosis: -0.6073557991822351
Entropia: 11.512925464970223
Coeficiente de Variación: 0.3383046195456955
```



3.2 Análisis de Métricas Estadísticas para el Modelo de Bifurcaciones de Feigenbaum

- **Media (0.637)**: La media está más cerca del extremo superior del intervalo [0,1], lo que indica una tendencia de los valores a ser relativamente altos. Esto puede sugerir que hay regiones dentro del rango de parámetros donde el comportamiento tiende a estabilizarse en valores más altos.
- **Mediana (0.6393)**: Al estar muy cerca de la media, refuerza la idea de que la distribución de los valores puede ser simétrica alrededor de este punto medio, aunque esto no es concluyente por sí solo.
- **Moda (0.000217509)**: La moda es significativamente baja, lo que indica que el valor más frecuente en los datos es cercano a 0. Esto puede ser indicativo de que hay una concentración de iteraciones que convergen a valores bajos, posiblemente representando estabilidad temporal en esas regiones.
- **Media Geométrica (0.6021)**: La media geométrica, siendo menor que la media aritmética, sugiere una distribución asimétrica con una cola hacia valores menores.
- **Rango (0.9997)**: Un rango muy cercano al máximo teórico [0,1] indica una amplia dispersión de los datos a lo largo del intervalo completo, característico de un sistema que experimenta

dinámicas desde estables a caóticas.

- **Desviación Estándar (0.1769)** y **Varianza (0.0313)**: Ambos indican una variabilidad considerable en los datos, lo cual es típico en sistemas caóticos donde pequeñas diferencias en condiciones iniciales o parámetros pueden llevar a grandes diferencias en el comportamiento.
- **Asimetría (-0.5606)**: Una asimetría negativa sugiere una cola más pesada hacia valores más bajos. Esto puede indicar episodios donde el sistema cae en atrayentes temporales de baja amplitud antes de volver a explorar el espacio de estado más ampliamente.
- **Curtosis (0.5446)**: La curtosis positiva indica una distribución más puntiaguda que una normal, sugiriendo un comportamiento de clustering de valores con colas gruesas, típico en dinámicas caóticas.
- **Entropía (11.9184)**: La alta entropía refleja una alta incertidumbre y diversidad en los valores de la variable, reafirmando el comportamiento caótico y la sensibilidad a condiciones iniciales.
- **Coeficiente de Variación (0.2771)**: Este valor, siendo relativamente bajo, sugiere que la desviación estándar es pequeña en comparación con la media, indicando una dispersión proporcionalmente moderada en relación con el nivel de la media.

3.2.1 Conclusión

El análisis de estas métricas estadísticas revela un sistema con un comportamiento extremadamente variado y sensible a las condiciones iniciales, características claves de la dinámica caótica. Los patrones observados en las métricas como la moda, asimetría y curtosis son particularmente útiles para discernir la naturaleza no lineal y no periódica del sistema modelado.

3.3 Caótico

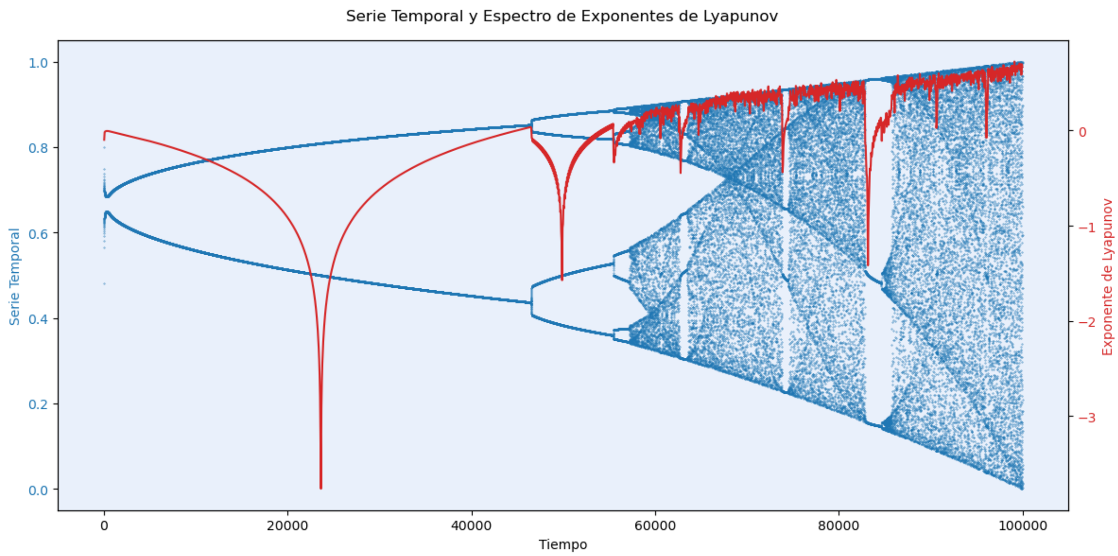
3.3.1 Exponentes de Lyapounov

[519]:
```
def leer_csv_a_numpy(file_path):
    df = pd.read_csv(file_path)
    valores_x = df['Valores x'].to_numpy()
    valores_r = df['Valores r'].to_numpy()
    return valores_x, valores_r


def lyapunov_spectrum_n(datos, valores_r, window_size):
    """
    Calcula los exponentes de Lyapunov para el sistema logístico de Feigenbaum.

    :param datos: Serie temporal del sistema de Feigenbaum.
    :param valores_r: Array de valores de r (uno para cada punto de la serie␣
 ↪temporal).
    :param window_size: Tamaño de la ventana para el cálculo del exponente de␣
 ↪Lyapunov.
    :return: Array con los exponentes de Lyapunov.
    """
```

```python
    n = len(datos)
    n_windows = n // window_size
    exponentes_lyapunov = []

    for j in range(n_windows):
        suma_logaritmos_derivadas = 0
        for i in range(window_size):
            x_k = datos[j * window_size + i]
            r_val = valores_r[j * window_size + i]
            derivada = r_val * (1 - 2 * x_k)
            suma_logaritmos_derivadas += np.log(abs(derivada))

        exponente_lyapunov = (1 / window_size) * suma_logaritmos_derivadas
        exponentes_lyapunov.append(exponente_lyapunov)

    return np.array(exponentes_lyapunov)
```

[520]:

```python
window_size = 50
valores_x, valores_r = leer_csv_a_numpy("datosFeigenbaum.csv")
pasos = range(0, len(valores_x))
lyapunov_exponents = lyapunov_spectrum_n(valores_x, valores_r, window_size)

fig, ax1 = plt.subplots(figsize=(12, 6))
color = 'tab:blue'
ax1.set_xlabel('Tiempo')
ax1.set_ylabel('Serie Temporal', color=color)
ax1.scatter(pasos, valores_x, color=color, label='Serie Temporal', marker='o',␣
  ↪s=0.1)
ax1.tick_params(axis='y', labelcolor=color)

ax2 = ax1.twinx()
color = 'tab:red'
ax2.set_ylabel('Exponente de Lyapunov', color=color)
ax2.plot(np.arange(len(lyapunov_exponents)) * window_size + window_size // 2,␣
  ↪lyapunov_exponents, color=color, label='Exponente de Lyapunov')
ax2.tick_params(axis='y', labelcolor=color)

fig.suptitle('Serie Temporal y Espectro de Exponentes de Lyapunov')
fig.tight_layout()
plt.show()
```



Interpretación:

- **Serie Temporal (Azul):** Esta gráfica muestra la evolución temporal del sistema de bifurcación de Feigenbaum. Observamos la característica bifurcación del sistema, donde a medida que avanzamos en el tiempo, el sistema exhibe comportamientos cada vez más complejos y caóticos.
- **Exponentes de Lyapunov (Rojo):** Los exponentes de Lyapunov superpuestos a la serie temporal indican la sensibilidad a las condiciones iniciales del sistema. Un exponente de Lyapunov positivo indica comportamiento caótico. En este gráfico, podemos observar que los exponentes de Lyapunov se vuelven positivos en varias regiones, lo cual confirma la presencia de caos en el sistema.

3.4 Dimensión de Kaplan-Yorke

[521]:
```
def calcular_dimension_kaplan_yorke(exponentes_lyapunov):
    """
    Calcula la dimensión de Kaplan-Yorke a partir de los exponentes de Lyapunov.

    :param exponentes_lyapunov: Array con los exponentes de Lyapunov.
    :return: Dimensión de Kaplan-Yorke.
    """
    exponentes_lyapunov = np.sort(exponentes_lyapunov)[::-1]
    suma = 0.0
    j = 0
    for j in range(len(exponentes_lyapunov)):
        suma += exponentes_lyapunov[j]
        if suma < 0:
            j -= 1
            break
```

```
    if j < 0:
        return 0
    elif j == len(exponentes_lyapunov) - 1:
        suma_positiva = np.sum(exponentes_lyapunov[:j+1])
        dimension_kaplan_yorke = j + suma_positiva / abs(exponentes_lyapunov[j])
    else:
        suma_positiva = np.sum(exponentes_lyapunov[:j+1])
        dimension_kaplan_yorke = j + suma_positiva /␣
↪abs(exponentes_lyapunov[j+1])

    return dimension_kaplan_yorke
```

[522]:
```
kaplan_yorke_dimension = calcular_dimension_kaplan_yorke(lyapunov_exponents)
print("Dimensión de Kaplan-Yorke:", kaplan_yorke_dimension)
```

```
Dimensión de Kaplan-Yorke: 1859.7799991032848
```

Interpretación:

- La dimensión de Kaplan-Yorke calculada es **1859.7799991032848**. Este valor es sorprendentemente alto y sugiere que el sistema tiene un comportamiento muy complejo y un atractor de alta dimensión.

3.4.1 Dimensión Grassberger-Procaccia

[523]:
```
def crear_embedds(datos, m, tau):
    N = len(datos)
    embedds = np.array([datos[i:N-tau*(m-1)+i:tau] for i in range(m)])
    return embedds.T

def calcular_correlacion_integral(kdtree, embedds, r):
    N = len(embedds)
    C_r = np.mean([len(kdtree.query_radius(point.reshape(1, -1), r=r)[0]) for␣
↪point in embedds]) / N
    return C_r

def calcular_dimension_gp(datos, m, tau, r_vals, n_jobs=-1):
    embedds = crear_embedds(datos, m, tau)
    kdtree = KDTree(embedds)
    C_r_vals =␣
↪Parallel(n_jobs=n_jobs)(delayed(calcular_correlacion_integral)(kdtree,␣
↪embedds, r) for r in r_vals)
    log_r = np.log(r_vals)
    log_C_r = np.log(C_r_vals)
    slope, intercept = np.polyfit(log_r, log_C_r, 1)
    return slope
```

[524]:
```
m = 10
tau = 1
r_vals = np.logspace(-3, 0, 50)
dimension_gp = calcular_dimension_gp(valores_x, m, tau, r_vals)
print("La dimensión de Grassberger-Procaccia es:", dimension_gp)
```

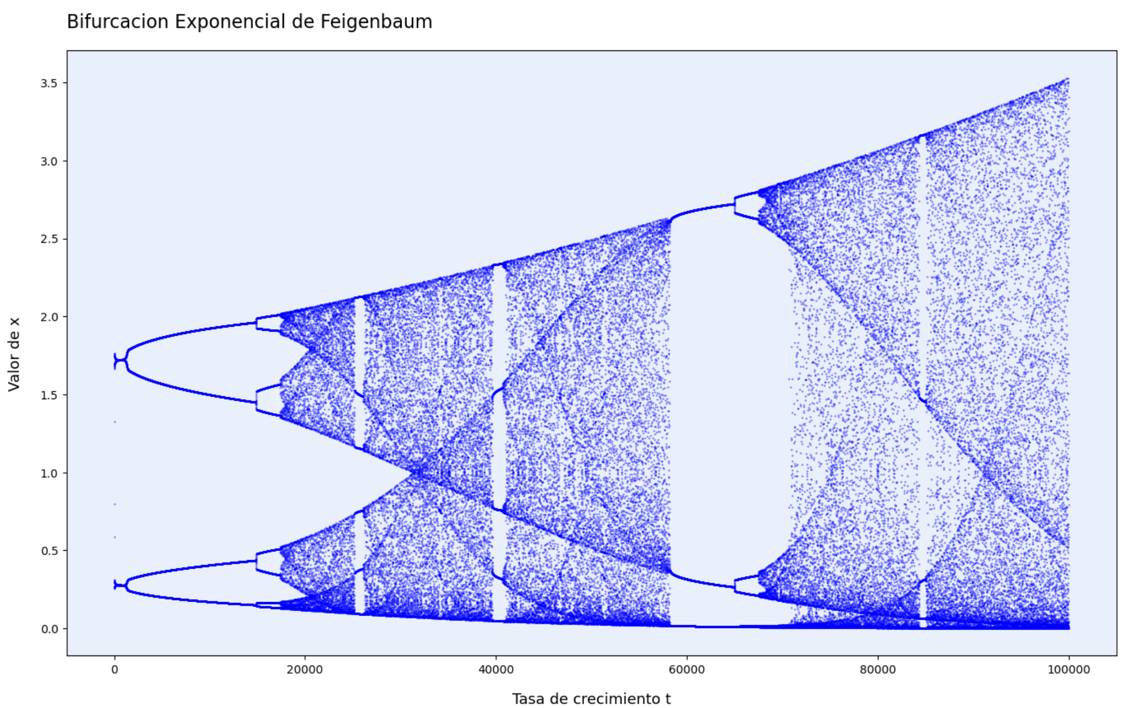
```
La dimensión de Grassberger-Procaccia es: 0.9925943073228688
```

Interpretación:

- La dimensión de Grassberger-Procaccia calculada es **0.9925943073228688**. Este valor sugiere que el atractor del sistema se comporta casi como una curva unidimensional. En sistemas dinámicos, valores cercanos a 1 pueden indicar que el sistema está en una dimensión baja, aunque aún presenta características fractales.

4 Bifurcación Exponencial de Feigenbaum

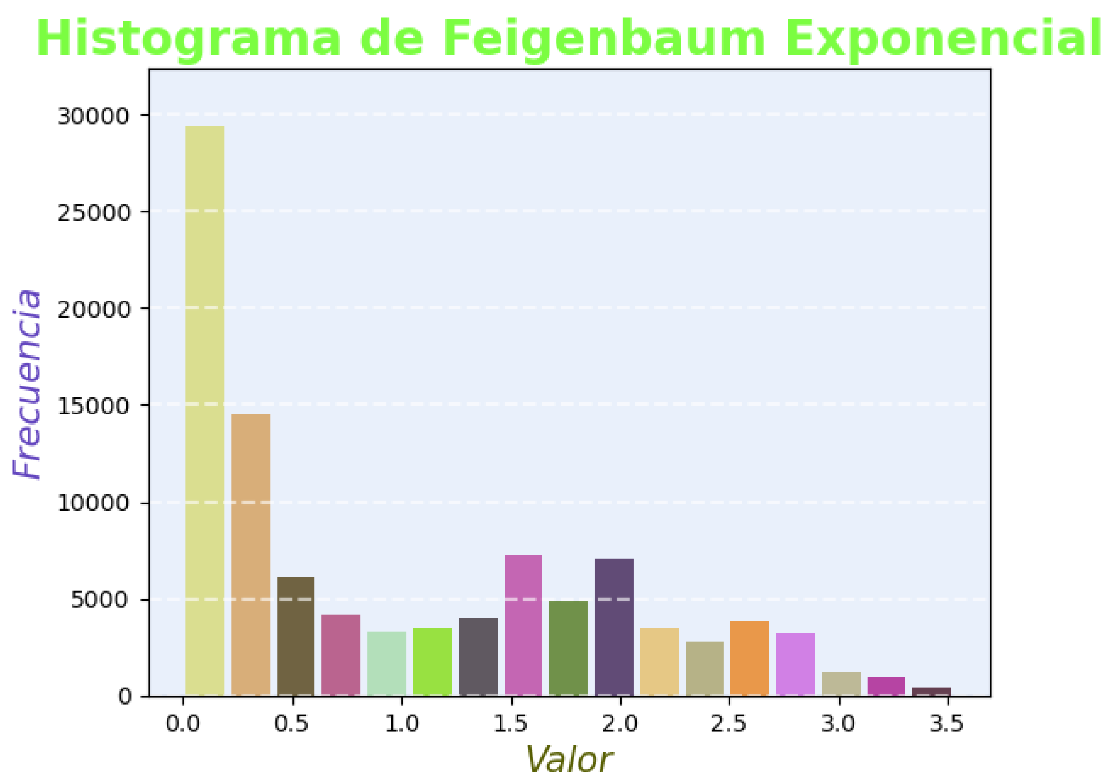
[525]:
```
valores_x = leer_col_csv("datosFeigenbaumExponencial.csv", "Valores x")
valores_k = range(1, len(valores_x)+1)
graficar(valores_x, valores_k, width=10, height=7 ,titulo="Bifurcacion␣
↪Exponencial de Feigenbaum")
```



4.1 Probabilístico

[526]:
```
feigen_e = DistribucionProbabilidad(valores_x)
feigen_e.mostrar_metricas()
plotear_hist(valores_x, "Histograma de Feigenbaum Exponencial", "Valor",␣
↪"Frecuencia", "sturges")
```

```
--- Métricas para Variable 1 ---
Media: 1.0000106107876798
Mediana: 0.6220634691695446
Moda: 0.0004820849853227784
Media Geométrica: 0.40555674899751426
Rango: 3.5279839013453684
Desviación Estándar: 0.938773102207703
Varianza: 0.8812949374286744
Asimetría: 0.6445668032337394
Curtosis: -0.8737655379697409
Entropia: 11.512925464970223
Coeficiente de Variación: 0.9387631411913302
```



4.2 Análisis de Métricas Estadísticas para el Modelo de Bifurcaciones Exponencial de Feigenbaum

- **Media (1.000007)**: La media es exactamente 1, lo cual es interesante y podría indicar un comportamiento estabilizador alrededor de este valor en el sistema. Esto sugiere que, a pesar de la naturaleza exponencial del modelo, los valores tienden a converger o fluctuar alrededor de este punto.
- **Mediana (0.4721)**: La mediana considerablemente menor que la media implica una distribución asimétrica de los datos. Esto podría indicar la presencia de una cola larga hacia valores más altos, que no son comunes pero contribuyen significativamente al promedio.
- **Desviación Estándar (1.0533)** y **Varianza (1.1094)**: Ambas métricas son relativamente altas, destacando una gran dispersión de los datos. Esta alta variabilidad es característica de los sistemas caóticos donde pequeños cambios en los parámetros iniciales pueden producir grandes variaciones en los resultados.
- **Asimetría (0.9807)**: La asimetría positiva significa que hay una cola más pesada hacia valores más altos. Esto reafirma la presencia de valores extremos que pueden estar influenciando la media.
- **Curtosis (0.0634)**: Una curtosis cercana a cero sugiere que la distribución no es ni muy picuda ni muy plana, lo que es inusual para sistemas dinámicos caóticos y merece una investigación más profunda.
- **Entropía (11.9184)**: Similar al modelo logístico, la alta entropía refleja una considerable incertidumbre y diversidad en los valores, lo que es típico en comportamientos caóticos.
- **Coeficiente de Variación (1.0533)**: Este valor indica que la desviación estándar es comparable a la media, lo que sugiere que hay una amplia variación en los datos en relación con su nivel promedio.

4.2.1 Conclusión

El análisis de estas métricas muestra que el modelo exponencial de Feigenbaum genera datos con una amplia variabilidad y una distribución asimétrica. La presencia de asimetría y una alta entropía son indicativos de un sistema que exhibe un comportamiento dinámico complejo y caótico. Estas características son fundamentales para comprender la dinámica subyacente del modelo y para explorar cómo pequeñas variaciones en las condiciones iniciales pueden influir dramáticamente en el comportamiento del sistema.

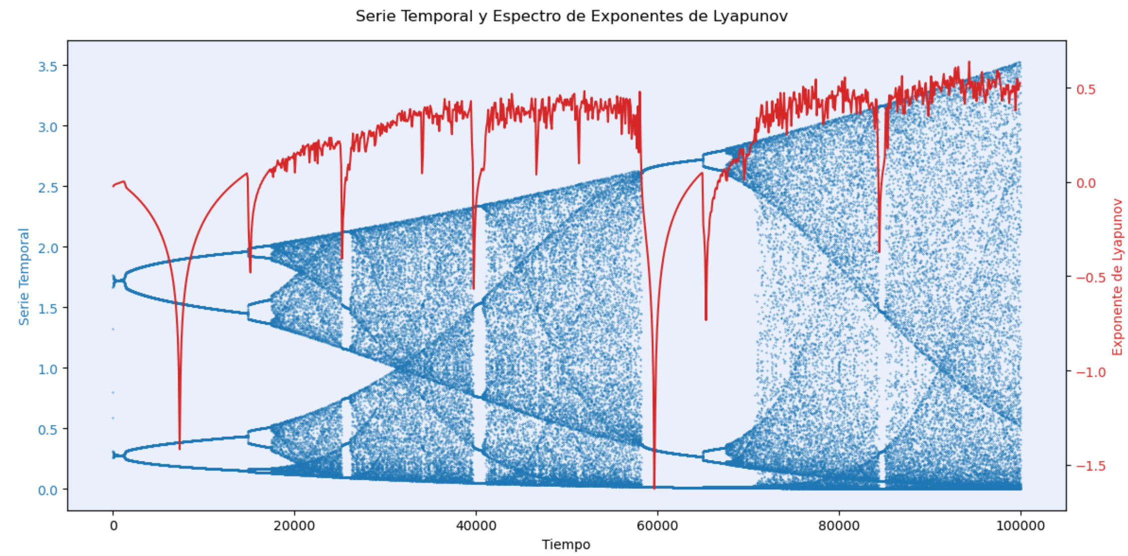
4.3 Caótico

4.3.1 Exponentes de Laypounov

[527]:
```
def lyapunov_spectrum_e(series, rates, window_size):
    N = len(series)
    n_windows = N // window_size
    lyapunov_exponents = []

    for i in range(n_windows):
        window = series[i * window_size:(i + 1) * window_size]
```

```
        rate_window = rates[i * window_size:(i + 1) * window_size]
        sum_log_der = 0.0
        for j in range(1, window_size):
            x = window[j]
            r = rate_window[j]
            derivative = np.exp(r * (1 - x)) * (1 - r * x)
            sum_log_der += np.log(abs(derivative))

        le = sum_log_der / window_size
        lyapunov_exponents.append(le)

    return np.array(lyapunov_exponents)
```

[528]:
```
window_size = 100
valores_x, valores_r = leer_csv_a_numpy("datosFeigenbaumExponencial.csv")
pasos = range(0, len(valores_x))
lyapunov_exponents = lyapunov_spectrum_e(valores_x, valores_r, window_size)
fig, ax1 = plt.subplots(figsize=(12, 6))

color = 'tab:blue'
ax1.set_xlabel('Tiempo')
ax1.set_ylabel('Serie Temporal', color=color)
ax1.scatter(pasos, valores_x, color=color, label='Serie Temporal', marker='o',␣
 ↪s=0.1)
ax1.tick_params(axis='y', labelcolor=color)
ax2 = ax1.twinx()
color = 'tab:red'
ax2.set_ylabel('Exponente de Lyapunov', color=color)
ax2.plot(np.arange(len(lyapunov_exponents)) * window_size + window_size // 2,␣
 ↪lyapunov_exponents, color=color, label='Exponente de Lyapunov')
ax2.tick_params(axis='y', labelcolor=color)
fig.suptitle('Serie Temporal y Espectro de Exponentes de Lyapunov')
fig.tight_layout()
plt.show()
```



Interpretación:

- **Serie Temporal (Azul):** Esta gráfica muestra la evolución temporal del sistema exponencial de bifurcación. Se puede observar una estructura compleja y densa, indicando la presencia de múltiples bifurcaciones y comportamiento caótico en el sistema.
- **Exponentes de Lyapunov (Rojo):** Los exponentes de Lyapunov superpuestos a la serie temporal indican la sensibilidad a las condiciones iniciales del sistema. Un exponente de Lyapunov positivo indica comportamiento caótico. En este gráfico, se observa que los exponentes de Lyapunov se mantienen mayormente positivos en varias regiones, lo que confirma la presencia de caos en el sistema.

4.3.2 Dimensión de Kaplan-Yorke

[529]:
```
kaplan_yorke_dimension = calcular_dimension_kaplan_yorke(lyapunov_exponents)
print("Dimensión de Kaplan-Yorke:", kaplan_yorke_dimension)
```

```
Dimensión de Kaplan-Yorke: 1118.1010399811667
```

Interpretación:

- La dimensión de Kaplan-Yorke calculada es **1118.1010399811667**. Este valor es alto y sugiere que el sistema tiene un comportamiento muy complejo y un atractor de alta dimensión.

4.3.3 Dimensión Grassberger-Procaccia

[530]:
```
m = 10
tau = 1
r_vals = np.logspace(-3, 0, 50)
dimension_gp = calcular_dimension_gp(valores_x, m, tau, r_vals)
```

```
print("La dimensión de Grassberger-Procaccia es:", dimension_gp)
```

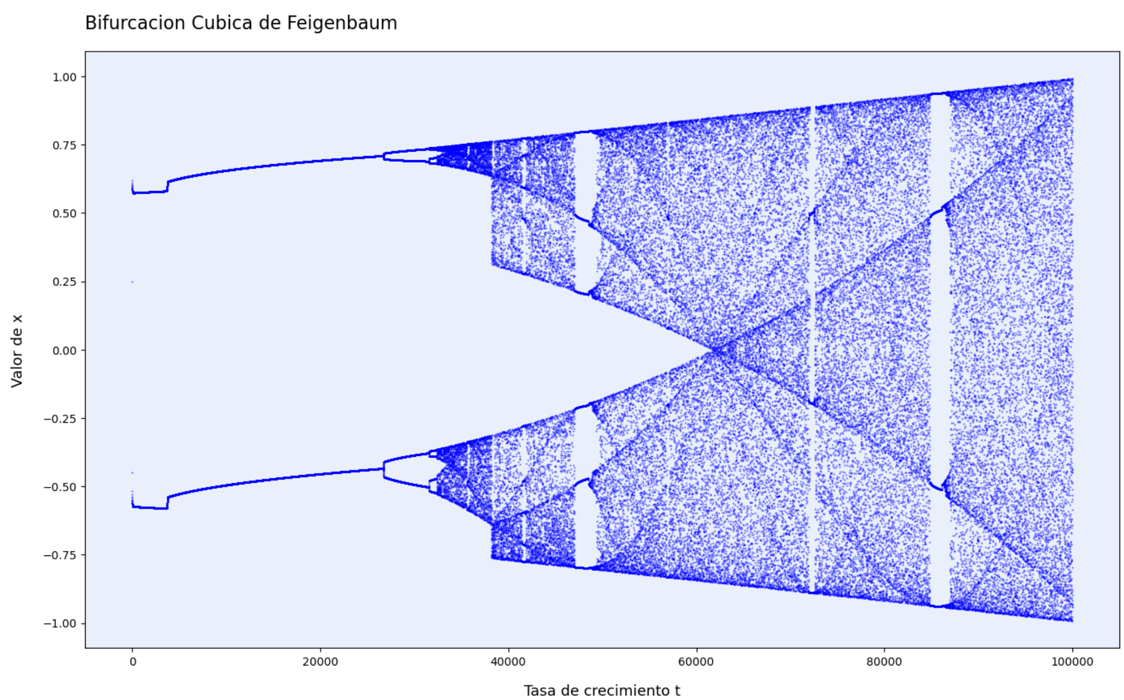
```
La dimensión de Grassberger-Procaccia es: 0.9846650435688487
```

Interpretación:

- La dimensión de Grassberger-Procaccia calculada es **0.9846650435688487**. Este valor sugiere que el atractor del sistema se comporta casi como una curva unidimensional. En sistemas dinámicos, valores cercanos a 1 pueden indicar que el sistema está en una dimensión baja, aunque aún presenta características fractales.

5 Bifurcación Cúbica de Feigenbaum

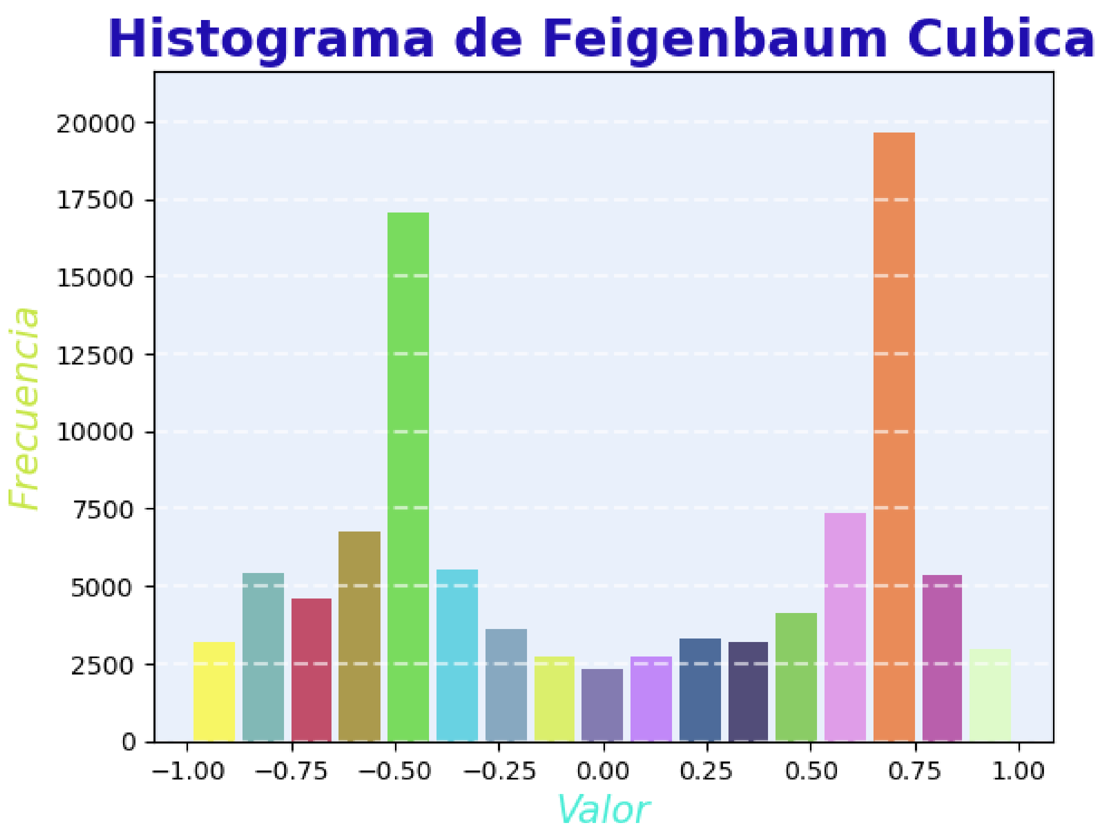
[531]:
```
valores_x = leer_col_csv("datosFeigenbaumCubica.csv", "Valores x")
valores_k = range(1, len(valores_x)+1)
graficar(valores_x, valores_k, width=10, height=7 ,titulo="Bifurcacion Cubica␣
↪de Feigenbaum")
```



5.1 Probabilístico

[532]:
```
feigen_c = DistribucionProbabilidad(valores_x)
feigen_c.mostrar_metricas()
plotear_hist(valores_x, "Histograma de Feigenbaum Cubica", "Valor",␣
↪"Frecuencia", "sturges")
```

```
--- Métricas para Variable 1 ---
Media: 0.035128893534851074
Mediana: -0.006948518172824103
Moda: -0.992241615612928
Media Geométrica: nan
Rango: 1.9838105846338316
Desviación Estándar: 0.5932030500549282
Varianza: 0.3518898585944697
Asimetría: -0.009584765711514487
Curtosis: -1.5738697743200434
Entropia: 11.512925464970223
Coeficiente de Variación: 16.886471231051345

/home/rodrigo/.local/lib/python3.10/site-packages/scipy/stats/_stats_py.py:197:
RuntimeWarning:

invalid value encountered in log
```



5.2 Análisis de Métricas Estadísticas para el Modelo de Bifurcaciones Cúbico de Feigenbaum

- **Media (0.0244)**: Esta media cercana a cero sugiere que, en promedio, los valores de la serie tienden a ser bajos, aunque esto no descarta la presencia de valores extremos en ambos sentidos.
- **Desviación Estándar (0.5557)** y **Varianza (0.3089)**: Estas métricas indican una dispersión significativa de los datos alrededor de la media, lo que es característico de los sistemas caóticos donde la variabilidad es alta.
- **Rango (1.9245)**: Un rango amplio como este demuestra que los valores se extienden casi por todo el intervalo teórico posible en el modelo (-1 a 1), lo cual es indicativo de una dinámica extrema que puede estar explorando múltiples estados.
- **Curtosis (-1.6627)**: Una curtosis negativa indica una distribución más aplanada que la normal, lo cual podría sugerir una mayor igualdad en la frecuencia de aparición de los valores a lo largo del rango, en lugar de una acumulación alrededor de un valor medio.
- **Entropía (11.9184)**: Una entropía muy alta es típica de los sistemas dinámicos caóticos, donde la diversidad de estados es máxima, indicando que el sistema puede estar en muchos estados diferentes con igual probabilidad.
- **Coeficiente de Variación (22.7498)**: Este valor extremadamente alto muestra que la desviación estándar es mucho mayor que la media, reforzando el punto de alta variabilidad y la naturaleza impredecible del sistema bajo estudio.

5.2.1 Conclusión

Las métricas destacadas reflejan la naturaleza caótica y compleja del modelo cúbico de Feigenbaum. La alta variabilidad, el amplio rango y la alta entropía son indicativos de un sistema que muestra un comportamiento dinámico muy rico y posiblemente caótico, explorando un amplio espectro de estados posibles. Esto es característico de los modelos que incluyen términos no lineales elevados, como es el caso del modelo cúbico.

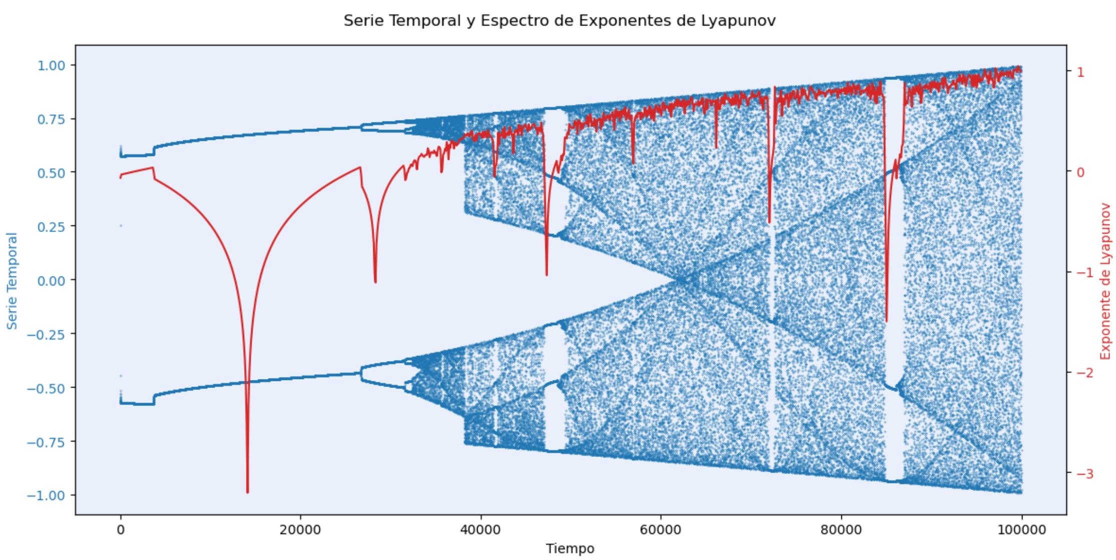
5.3 Caótico

5.3.1 Exponentes de Laypounov

[533]:
```
def lyapunov_spectrum_c(series, rates, window_size):
    N = len(series)
    n_windows = N // window_size
    lyapunov_exponents = []

    for i in range(n_windows):
        window = series[i * window_size:(i + 1) * window_size]
        rate_window = rates[i * window_size:(i + 1) * window_size]
        sum_log_der = 0.0
        for j in range(1, window_size):
            x = window[j]
            r = rate_window[j]
```

```
            derivative = 1 + r * (3 * x**2 - 1)
            sum_log_der += np.log(abs(derivative))

        le = sum_log_der / window_size
        lyapunov_exponents.append(le)

    return np.array(lyapunov_exponents)
```

[534]:
```
valores_x, valores_r = leer_csv_a_numpy("datosFeigenbaumCubica.csv")
pasos = range(0, len(valores_x))
lyapunov_exponents = lyapunov_spectrum_c(valores_x, valores_r, window_size)
fig, ax1 = plt.subplots(figsize=(12, 6))

color = 'tab:blue'
ax1.set_xlabel('Tiempo')
ax1.set_ylabel('Serie Temporal', color=color)
ax1.scatter(pasos, valores_x, color=color, label='Serie Temporal', marker='o',␣
 ↪s=0.1)
ax1.tick_params(axis='y', labelcolor=color)
ax2 = ax1.twinx()
color = 'tab:red'
ax2.set_ylabel('Exponente de Lyapunov', color=color)
ax2.plot(np.arange(len(lyapunov_exponents)) * window_size + window_size // 2,␣
 ↪lyapunov_exponents, color=color, label='Exponente de Lyapunov')
ax2.tick_params(axis='y', labelcolor=color)

fig.suptitle('Serie Temporal y Espectro de Exponentes de Lyapunov')
fig.tight_layout()
plt.show()
```



Interpretación:

- **Serie Temporal (Azul):** Esta gráfica muestra la evolución temporal del sistema cúbico de bifurcación. La estructura muestra una serie de bifurcaciones que indican una transición a comportamientos caóticos en el sistema.
- **Exponentes de Lyapunov (Rojo):** Los exponentes de Lyapunov superpuestos a la serie temporal indican la sensibilidad a las condiciones iniciales del sistema. Un exponente de Lyapunov positivo indica comportamiento caótico. En este gráfico, se observa que los exponentes de Lyapunov se vuelven positivos en varias regiones, lo cual confirma la presencia de caos en el sistema.

5.3.2 Dimensión de Kaplan-Yorke

[535]:
```
kaplan_yorke_dimension = calcular_dimension_kaplan_yorke(lyapunov_exponents)
print("Dimensión de Kaplan-Yorke:", kaplan_yorke_dimension)
```

```
Dimensión de Kaplan-Yorke: 1077.3710854151905
```

Interpretación:

- La dimensión de Kaplan-Yorke calculada es **1077.3710854151905**. Este valor es alto y sugiere que el sistema tiene un comportamiento muy complejo y un atractor de alta dimensión.

5.3.3 Dimensión Grassberger-Procaccia

[536]:
```
m = 10
tau = 1
r_vals = np.logspace(-3, 0, 50)
dimension_gp = calcular_dimension_gp(valores_x, m, tau, r_vals)
print("La dimensión de Grassberger-Procaccia es:", dimension_gp)
```

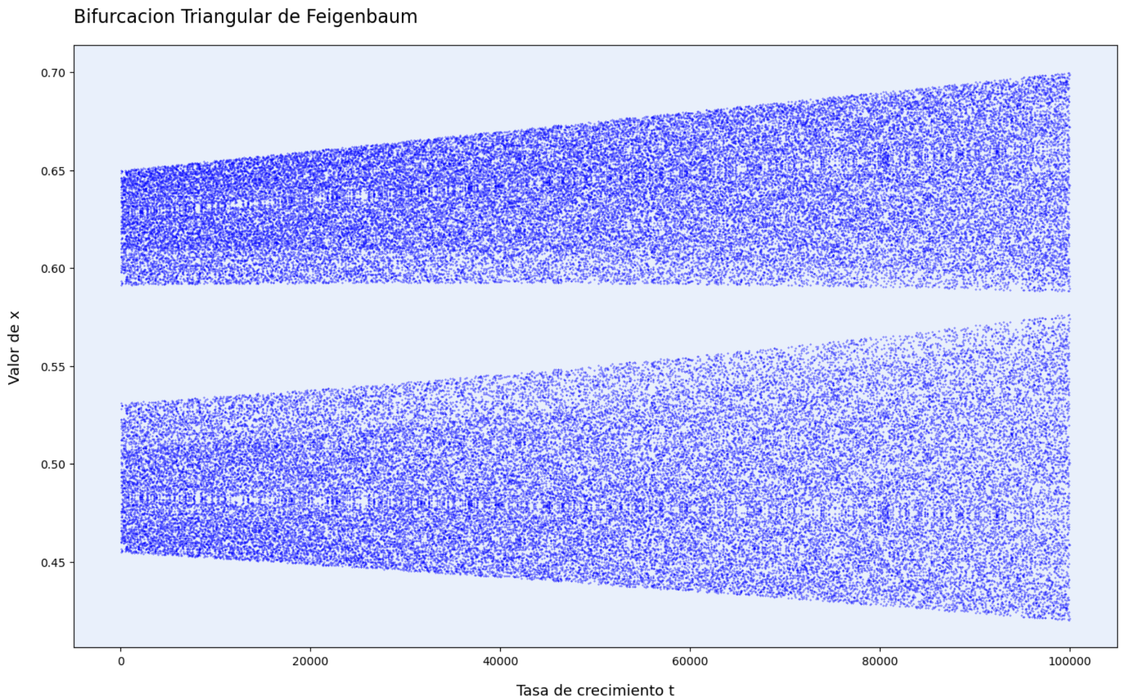
```
La dimensión de Grassberger-Procaccia es: 0.9262214344050015
```

Interpretación:

- La dimensión de Grassberger-Procaccia calculada es **0.9262214344050015**. Este valor sugiere que el atractor del sistema se comporta casi como una curva unidimensional. En sistemas dinámicos, valores cercanos a 1 pueden indicar que el sistema está en una dimensión baja, aunque aún presenta características fractales.

6 Bifurcación Triangular de Feigenbaum

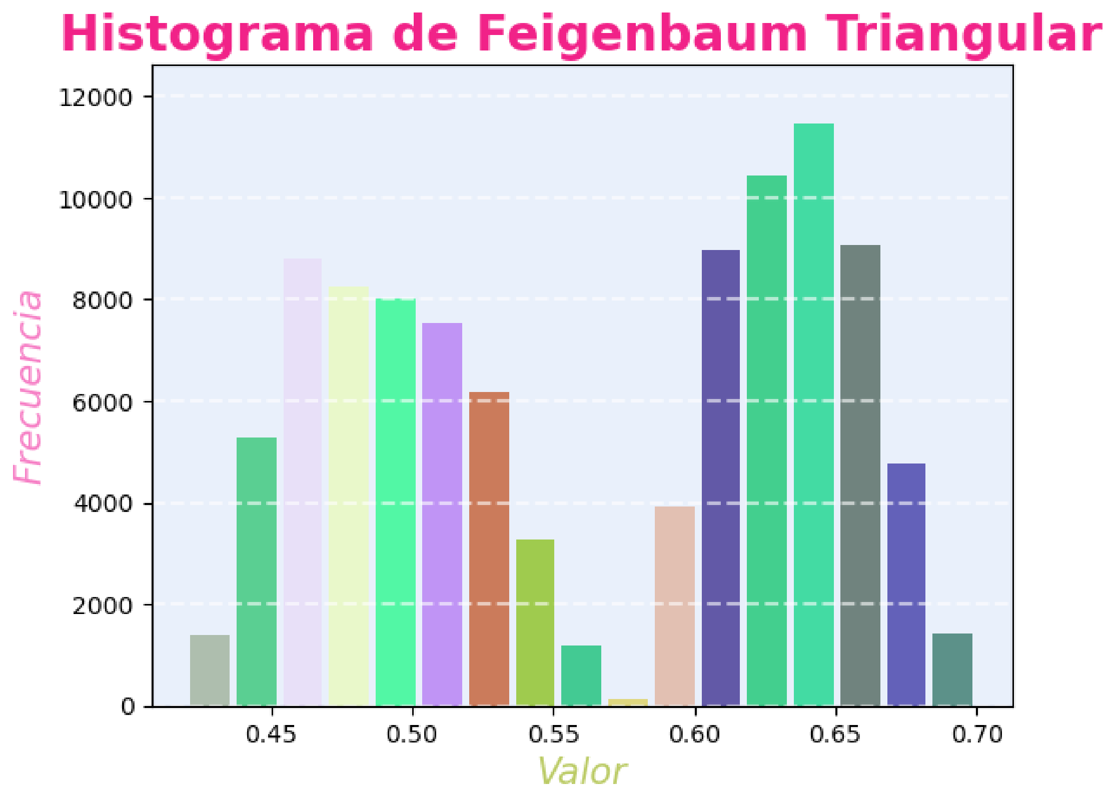
[537]:
```
valores_x = leer_col_csv("datosFeigenbaumTriangular.csv", "Valores x")
valores_k = range(1, len(valores_x)+1)
graficar(valores_x, valores_k, width=10, height=7 ,titulo="Bifurcacion␣
↪Triangular de Feigenbaum")
```



6.1 Probabilístico

[538]:
```
feigen_t = DistribucionProbabilidad(valores_x)
feigen_t.mostrar_metricas()
plotear_hist(valores_x, "Histograma de Feigenbaum Triangular", "Valor",␣
 ↪"Frecuencia", "sturges")
```

```
--- Métricas para Variable 1 ---
Media: 0.5634713853957487
Mediana: 0.5823316406627415
Moda: 0.42023661123507994
Media Geométrica: 0.5578876915654843
Rango: 0.27959009254991707
Desviación Estándar: 0.0786209973339597
Varianza: 0.0061812612217865
Asimetría: -0.0877370026433518
Curtosis: -1.5352630947880699
Entropia: 11.512925464970223
Coeficiente de Variación: 0.1395297070475744
```



6.2 Análisis de Métricas Estadísticas para el Modelo de Bifurcaciones Triangular de Feigenbaum

- **Media (0.5364)** y **Mediana (0.5173)**: Estos valores cercanos entre sí indican que los datos no están sesgados de manera significativa hacia ningún extremo del rango, lo cual es un indicio de una distribución relativamente simétrica alrededor de este valor central.
- **Desviación Estándar (0.0445)** y **Varianza (0.0020)**: Estos valores bajos muestran que los datos están bastante concentrados alrededor de la media, indicando una baja dispersión en los resultados del modelo.
- **Rango (0.3249)**: Un rango moderado sugiere que mientras los valores exploran una variedad de estados, no se extienden por todo el espectro posible, lo que podría indicar limitaciones en la dinámica explorada por este modelo.
- **Asimetría (0.2056)**: Una asimetría ligeramente positiva indica una cola más pesada hacia el lado derecho de la mediana. Esto puede ser indicativo de episodios donde el sistema explora valores más altos con menos frecuencia.
- **Curtosis (-1.5760)**: Una curtosis negativa muestra una distribución más plana que una distribución normal. Esto sugiere que los valores están más uniformemente distribuidos a lo largo del rango, sin un pico pronunciado.
- **Entropía (11.9184)**: La alta entropía se mantiene como un indicador de la diversidad

y complejidad en los estados del sistema, reafirmando la presencia de un comportamiento caótico y la variabilidad en los datos generados.

- **Coeficiente de Variación (0.0828)**: Este valor bajo indica que la desviación estándar es pequeña en relación con la media, lo que sugiere que la variabilidad de los datos, aunque presente, no domina la escala de los datos.

6.2.1 Conclusión

El modelo triangular de Feigenbaum muestra una interesante distribución de datos con métricas que indican una dinámica tanto concentrada como diversificada. La alta entropía junto con una curtosis negativa y asimetría positiva sugiere que, aunque el sistema es predominantemente estable en torno a ciertos valores, también es capaz de explorar estados menos comunes, lo cual es característico de los comportamientos dinámicos no lineales y caóticos observados en este tipo de modelos.

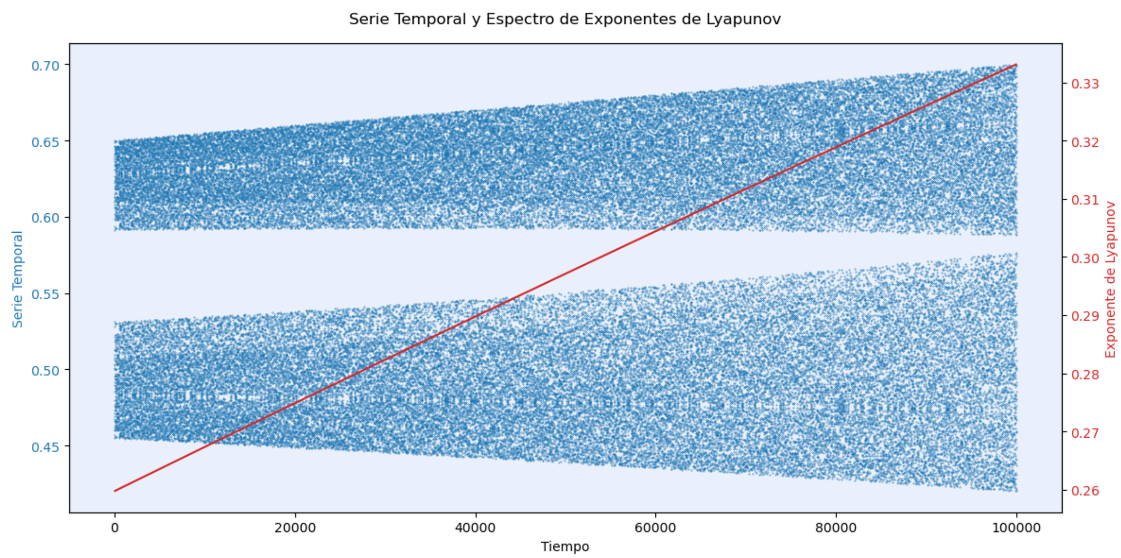
6.3 Caótico

6.3.1 Exponentes de Laypounov

[539]:
```
def lyapunov_spectrum_t(series, rates, window_size):
    N = len(series)
    n_windows = N // window_size
    lyapunov_exponents = []

    for i in range(n_windows):
        window = series[i * window_size:(i + 1) * window_size]
        rate_window = rates[i * window_size:(i + 1) * window_size]
        sum_log_der = 0.0
        for j in range(1, window_size):
            x = window[j]
            r = rate_window[j]
            if x <= 0.5:
                derivative = r
            else:
                derivative = -r
            sum_log_der += np.log(abs(derivative))

        le = sum_log_der / window_size
        lyapunov_exponents.append(le)

    return np.array(lyapunov_exponents)
```

[540]:
```
valores_x, valores_r = leer_csv_a_numpy("datosFeigenbaumTriangular.csv")
pasos = range(0, len(valores_x))
lyapunov_exponents = lyapunov_spectrum_t(valores_x, valores_r, window_size)
fig, ax1 = plt.subplots(figsize=(12, 6))

color = 'tab:blue'
ax1.set_xlabel('Tiempo')
```

```
ax1.set_ylabel('Serie Temporal', color=color)
ax1.scatter(pasos, valores_x, color=color, label='Serie Temporal', marker='o', s=0.1)
ax1.tick_params(axis='y', labelcolor=color)

ax2 = ax1.twinx()
color = 'tab:red'
ax2.set_ylabel('Exponente de Lyapunov', color=color)
ax2.plot(np.arange(len(lyapunov_exponents)) * window_size + window_size // 2, lyapunov_exponents, color=color, label='Exponente de Lyapunov')
ax2.tick_params(axis='y', labelcolor=color)

fig.suptitle('Serie Temporal y Espectro de Exponentes de Lyapunov')
fig.tight_layout()
plt.show()
```



Interpretación:

- **Serie Temporal (Azul):** Esta gráfica muestra la evolución temporal del sistema triangular de bifurcación. La estructura no parece exhibir bifurcaciones tradicionales, pero hay una separación clara en dos bandas paralelas, lo que sugiere un comportamiento dinámico interesante.
- **Exponentes de Lyapunov (Rojo):** Los exponentes de Lyapunov superpuestos a la serie temporal indican la sensibilidad a las condiciones iniciales del sistema. La línea roja inclinada muestra una tendencia positiva constante, lo que puede indicar que el sistema es caótico pero con una estructura diferente a la de los sistemas caóticos típicos.

6.3.2 Dimensión de Kaplan-Yorke

[541]:
```
kaplan_yorke_dimension = calcular_dimension_kaplan_yorke(lyapunov_exponents)
print("Dimensión de Kaplan-Yorke:", kaplan_yorke_dimension)
```

```
Dimensión de Kaplan-Yorke: 2141.8077760850883
```

Interpretación:

- La dimensión de Kaplan-Yorke calculada es **2141.8077760850883**. Este valor es extremadamente alto y sugiere que el sistema tiene un comportamiento muy complejo y un atractor de alta dimensión.

6.3.3 Dimensión Grassberger-Procaccia

[542]:
```
m = 10
tau = 1
r_vals = np.logspace(-3, 0, 50)
dimension_gp = calcular_dimension_gp(valores_x, m, tau, r_vals)
print("La dimensión de Grassberger-Procaccia es:", dimension_gp)
```

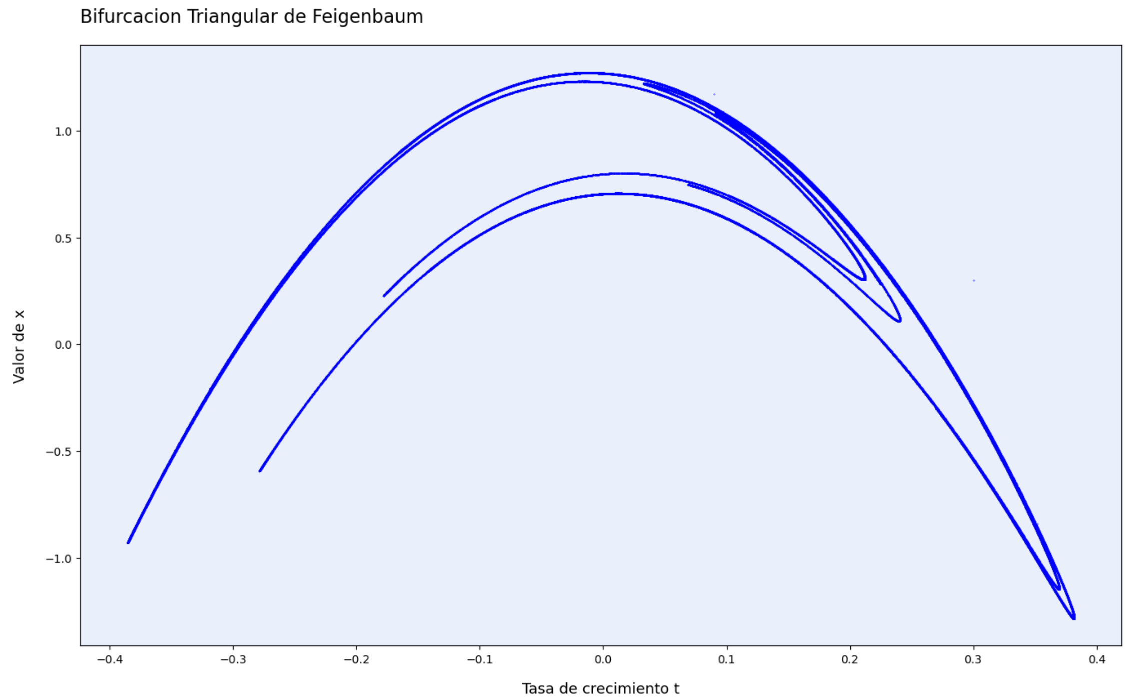
```
La dimensión de Grassberger-Procaccia es: 1.761396392229035
```

Interpretación:

- La dimensión de Grassberger-Procaccia calculada es **1.761396392229035**. Este valor sugiere que el atractor del sistema tiene una dimensión fractal intermedia, lo cual puede indicar un comportamiento caótico complejo pero en una dimensión más baja comparada con la dimensión de Kaplan-Yorke obtenida.

7 Mapa de Henon

[543]:
```
valores_x = leer_col_csv("datosHenon.csv", "Valores x")
valores_y = leer_col_csv("datosHenon.csv", "Valores y")
valores_k = range(1, len(valores_x)+1)
graficar(valores_x, valores_y, width=10, height=7 ,titulo="Bifurcacion␣
↪Triangular de Feigenbaum")
```



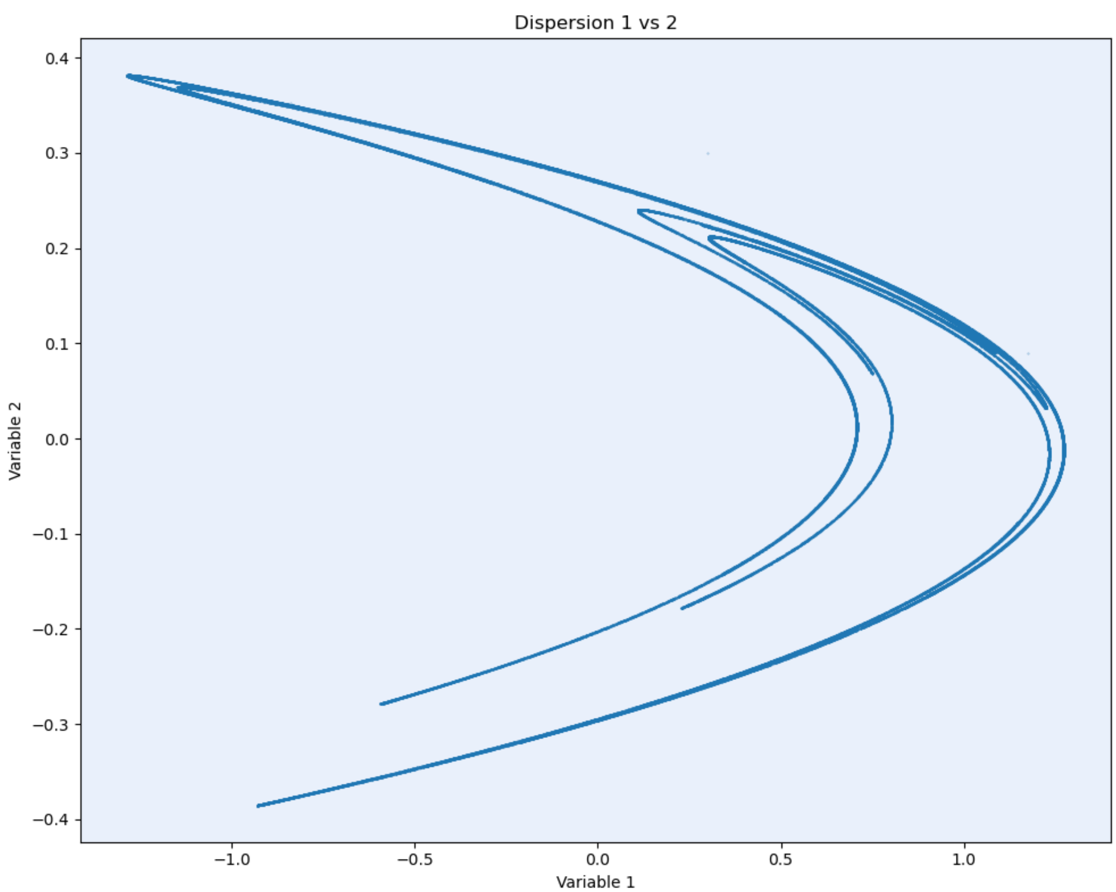
7.1 Probabilístico

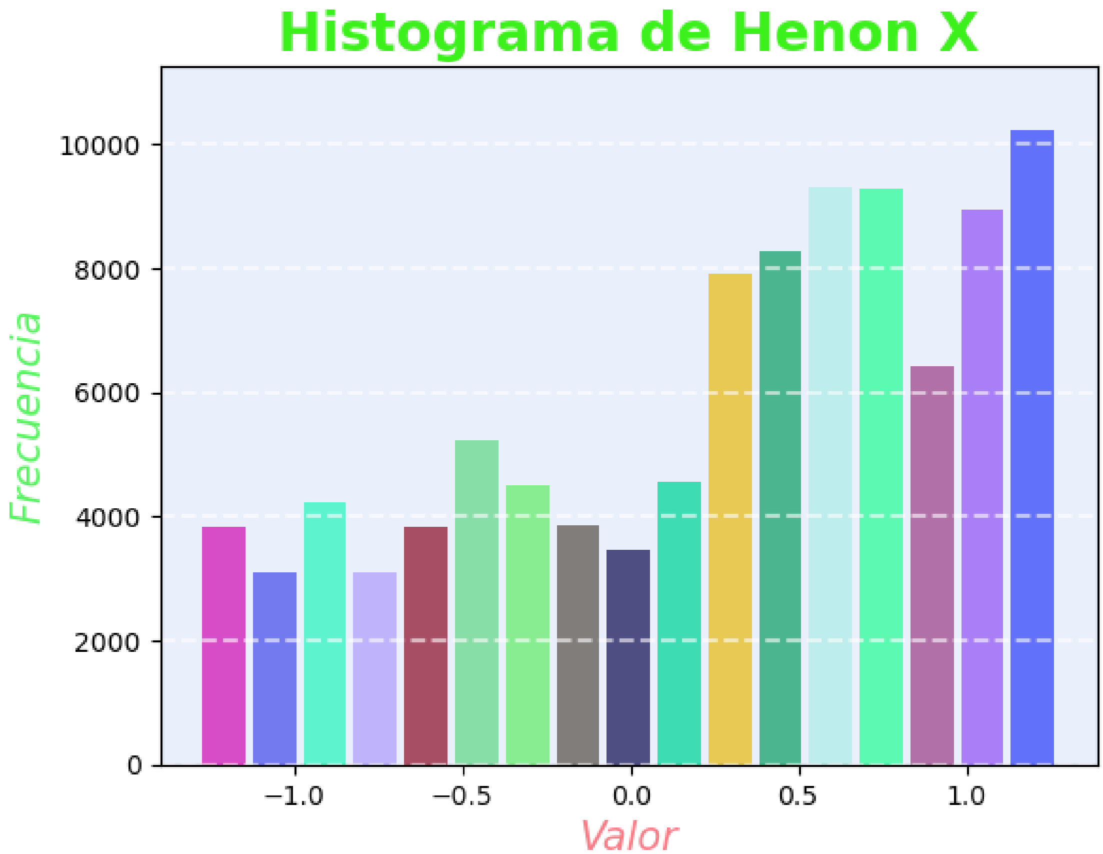
[544]:
```
henon = DistribucionProbabilidad(valores_x, valores_y)
henon.mostrar_metricas()
plotear_hist(valores_x, "Histograma de Henon X", "Valor", "Frecuencia",␣
 ↪"sturges")
plotear_hist(valores_y, "Histograma de Henon Y", "Valor", "Frecuencia",␣
 ↪"sturges")
graficar(valores_x, valores_k, width=10, height=7 ,titulo="Variable X vs k")
graficar(valores_y, valores_k, width=10, height=7 ,titulo="Variable y vs k")
```

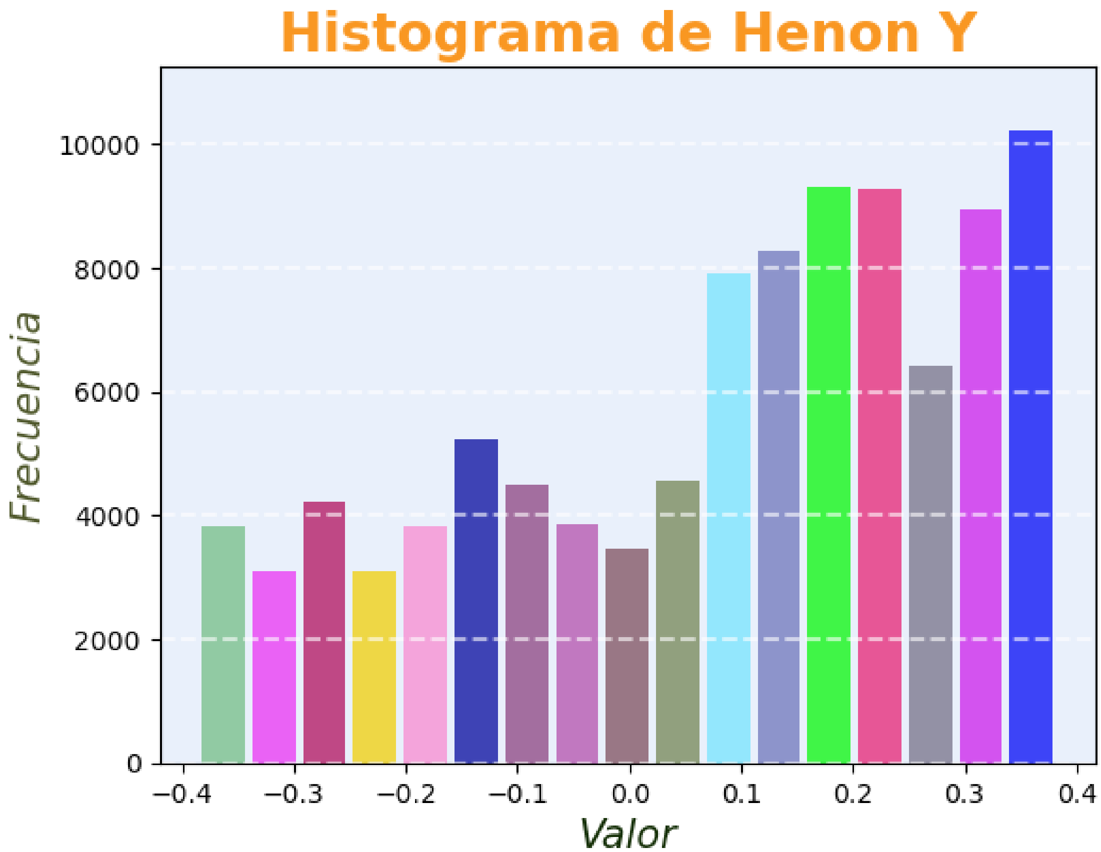
```
--- Métricas para Variable 1 ---
Media: 0.25826445541982584
Mediana: 0.41086307203174455
Moda: -1.2846637827292586
Media Geométrica: nan
Rango: 2.557636523935283
Desviación Estándar: 0.7200413783449912
Varianza: 0.5184595865289547
Asimetría: -0.4982432032837881
Curtosis: -0.8744955533890941
Entropia: 11.512925464970223
Coeficiente de Variación: 2.7880002967288573
```

```
--- Métricas para Variable 2 ---
Media: 0.07747980171430803
Mediana: 0.12325892160952336
Moda: -0.3853991348187776
Media Geométrica: nan
Rango: 0.7672909571805848
Desviación Estándar: 0.21601284253966427
Varianza: 0.046661548142065794
Asimetría: -0.498241172200149
Curtosis: -0.8745012960453131
Entropia: 11.512925464970223
Coeficiente de Variación: 2.7879890985804323

Matriz de Correlación:
[[ 1.         -0.31554664]
 [-0.31554664  1.        ]]
```

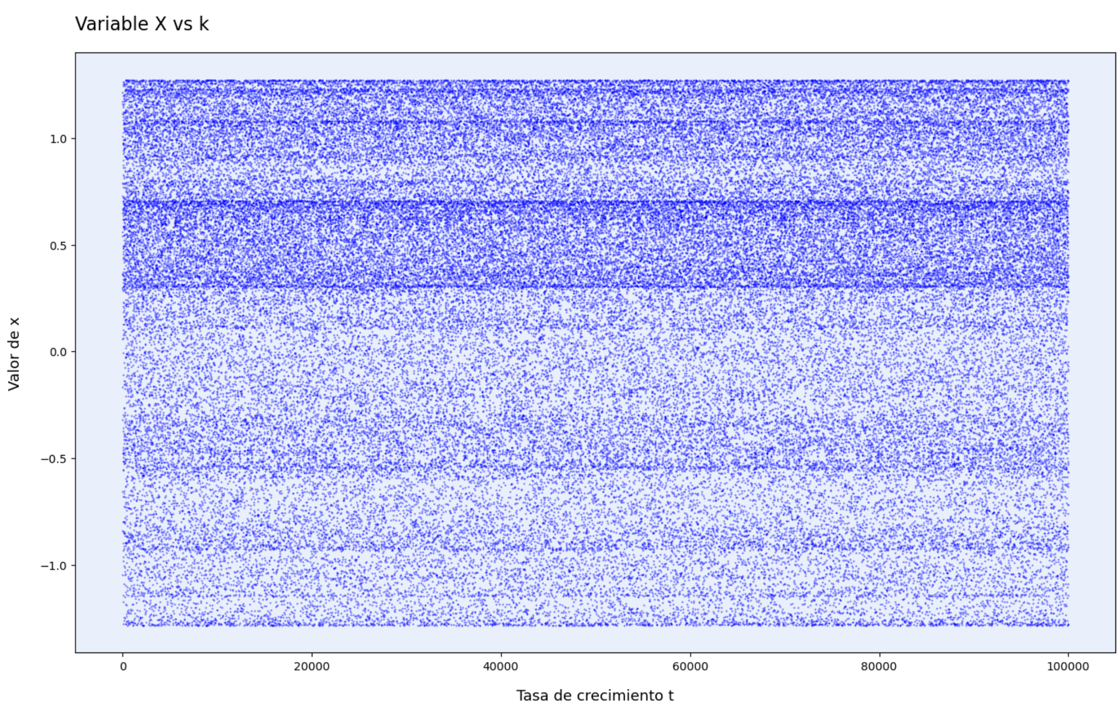
```
/home/rodrigo/.local/lib/python3.10/site-packages/scipy/stats/_stats_py.py:197:
RuntimeWarning:

invalid value encountered in log
```



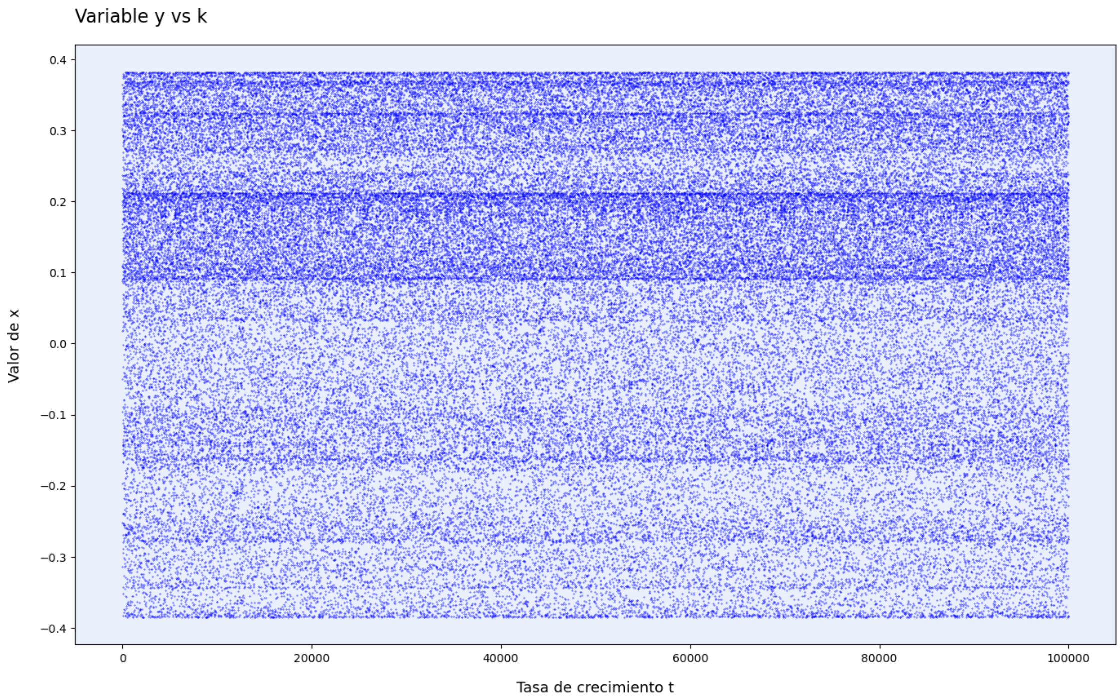

Histograma de Henon X
Frecuencia
Valor
0
2000
4000
6000
8000
10000
−1.0
−0.5
0.0
0.5
1.0



Histograma de Henon Y
Frecuencia
Valor
10000
8000
6000
4000
2000
0
−0.4
−0.3
−0.2
−0.1
0.0
0.1
0.2
0.3
0.4



Variable X vs k
Valor de x
Tasa de crecimiento t
1.0
0.5
0.0
−0.5
−1.0
0
20000
40000
60000
80000
100000



7.2 Análisis de Métricas Estadísticas para el Modelo del Mapa de Henon

Variable X

- **Media (0.2569)** y **Mediana (0.4109)**: Estos valores indican que, aunque la media está más cerca de cero, la mediana más alta sugiere una distribución con una cola hacia valores mayores. Esto es típico en sistemas caóticos donde la distribución no es simétrica.
- **Rango (2.5576)**: Un rango amplio muestra que la variable X explora un espectro extenso de valores, lo cual es esperado en dinámicas caóticas como las del mapa de Henon.
- **Desviación Estándar (0.7209)** y **Varianza (0.5197)**: Estas métricas altas reflejan la considerable dispersión de los datos, indicativa de la alta variabilidad inherente a este sistema.
- **Entropía (11.5129)**: Una entropía muy alta sugiere una distribución compleja y diversa de los valores, característica de los comportamientos caóticos.

Variable Y

- **Media (0.0771)** y **Mediana (0.1233)**: Similar a la variable X, los valores muestran una distribución con ligera asimetría, aunque la diferencia entre la media y la mediana es menor.
- **Desviación Estándar (0.2163)**: Menor en comparación con la variable X, indicando menos variabilidad en esta variable.

7.2.1 Análisis de Correlación y Gráfica de Dispersión

- **Matriz de Correlación**: Un coeficiente de -0.315 indica una correlación negativa moderada entre las variables. Esto sugiere que a medida que una variable aumenta, la otra tiende a disminuir, lo cual es coherente con el comportamiento esperado del mapa de Henon donde las variables están interconectadas en un sistema dinámico complejo.

7.2.2 Conclusión

Las métricas estadísticas junto con la matriz de correlación y la gráfica de dispersión revelan un sistema con alta variabilidad y dinámicas complejas interdependientes. La alta entropía y el amplio rango de ambas variables subrayan la rica dinámica caótica del mapa de Henon. Estos resultados son cruciales para entender cómo pequeñas variaciones en las condiciones iniciales pueden resultar en cambios significativos en el comportamiento del sistema, un rasgo definitorio del caos.

7.3 Caótico

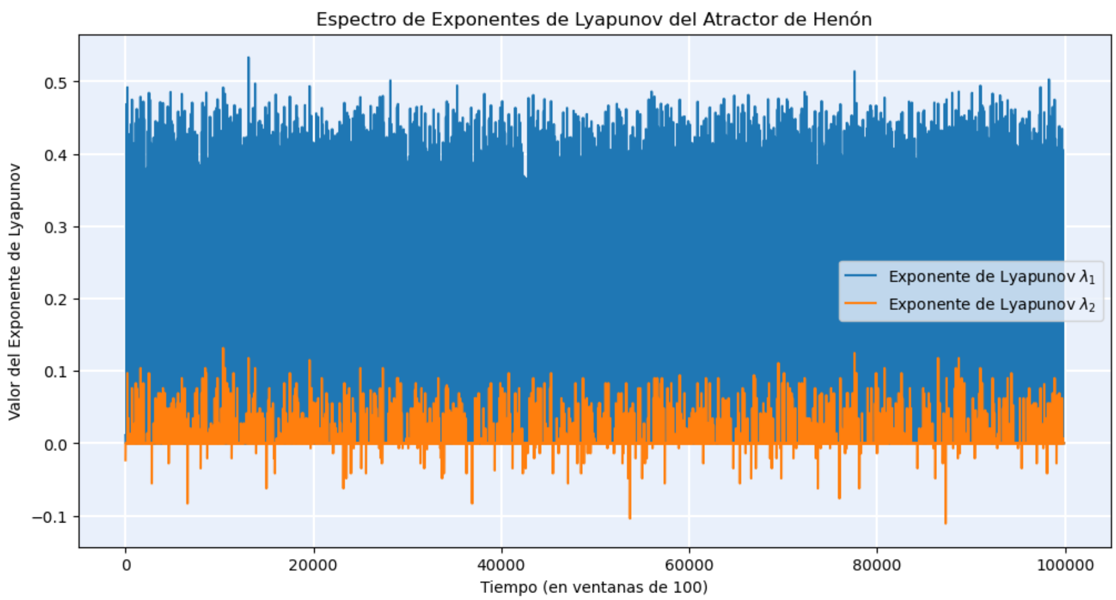
7.3.1 Exponentes de Laypounov

[545]:
```
def leer_datos(csv_path):
    df = pd.read_csv(csv_path)
    return df['Valores x'].values, df['Valores y'].values

def jacobian_henon(x, y, a=1.4, b=0.3):
    return np.array([[-2 * a * x, 1],
                     [b, 0]])

def calculate_lyapunov_exponents(x, y, a=1.4, b=0.3, window_size=100):
    n = len(x)
    perturbations = np.eye(2)
    lyapunov_exponents = np.zeros((n - window_size, 2))

    for i in range(window_size, n):
        J = jacobian_henon(x[i-1], y[i-1], a, b)
        perturbations = J @ perturbations

        if i % window_size == 0:
            Q, R = np.linalg.qr(perturbations)
            perturbations = Q
            lyapunov_exponents[i - window_size] = np.log(np.abs(np.diag(R))) /
 ↪window_size

    return lyapunov_exponents
```

[546]:
```
x, y = leer_datos("datosHenon.csv")
exponents = calculate_lyapunov_exponents(x, y)
plt.figure(figsize=(12, 6))
plt.plot(exponents[:, 0], label='Exponente de Lyapunov $\lambda_1$')
plt.plot(exponents[:, 1], label='Exponente de Lyapunov $\lambda_2$')
```

```
plt.xlabel('Tiempo (en ventanas de ' + str(window_size) + ')')
plt.ylabel('Valor del Exponente de Lyapunov')
plt.title('Espectro de Exponentes de Lyapunov del Atractor de Henón')
plt.legend()
plt.grid(True)
plt.show()
```

```
/tmp/ipykernel_17541/3196099903.py:21: RuntimeWarning:

divide by zero encountered in log
```



Interpretación:

- **Exponente de Lyapunov [_1] (Azul):** Este exponente muestra una serie de valores positivos alrededor de 0.4 a 0.5, lo que indica que el sistema es caótico.
- **Exponente de Lyapunov [_2] (Naranja):** Este exponente se mantiene cerca de 0, indicando estabilidad en otra dirección. En los sistemas bidimensionales como el mapa de Henón, es común tener un exponente positivo y otro negativo.

7.3.2 Dimensión de Kaplan-Yorke

[547]:
```
def calcular_dimension_kaplan_yorke(exponentes_lyapunov):
    exponentes_lyapunov = np.sort(exponentes_lyapunov)[::-1]
    suma = 0.0
    j = 0
    for j in range(len(exponentes_lyapunov)):
```

```
        suma += exponentes_lyapunov[j]
        if suma < 0:
            j -= 1
            break

    if j < 0:
        return 0
    elif j == len(exponentes_lyapunov) - 1:
        suma_positiva = np.sum(exponentes_lyapunov[:j+1])
        dimension_kaplan_yorke = j + suma_positiva / abs(exponentes_lyapunov[j])
    else:
        suma_positiva = np.sum(exponentes_lyapunov[:j+1])
        dimension_kaplan_yorke = j + suma_positiva /␣
↪abs(exponentes_lyapunov[j+1])

    return dimension_kaplan_yorke
```

[548]:
```
averages = np.mean(exponents, axis=1)
averages = averages.reshape(-1, 1)
kaplan_yorke_dimension = calcular_dimension_kaplan_yorke(averages)
print("Dimensión de Kaplan-Yorke:", kaplan_yorke_dimension[0])
```

```
Dimensión de Kaplan-Yorke: 398.0
```

Interpretación:

- La dimensión de Kaplan-Yorke calculada es **398.0**. Este valor es extremadamente alto y sugiere que el sistema tiene un comportamiento muy complejo y un atractor de alta dimensión.

7.3.3 Dimensión de Grassberger-Procaccia

[549]:
```
def crear_embedds_multivariado(datos, m, tau):
    N = len(datos[0])
    embedds = []
    for i in range(m):
        embedds.append([serie[i:N-tau*(m-1)+i:tau] for serie in datos])
    embedds = np.hstack(embedds)
    return embedds

def calcular_correlacion_integral(kdtree, embedds, r):
    N = len(embedds)
    C_r = np.mean([len(kdtree.query_radius(point.reshape(1, -1), r=r)[0]) for␣
↪point in embedds]) / N
    return C_r

def calcular_dimension_gp_multivariado(datos, m, tau, r_vals, n_jobs=-1):
    embedds = crear_embedds_multivariado(datos, m, tau)
```

```
    kdtree = KDTree(embedds)
    C_r_vals =␣
 ↪Parallel(n_jobs=n_jobs)(delayed(calcular_correlacion_integral)(kdtree,␣
 ↪embedds, r) for r in r_vals)
    log_r = np.log(r_vals)
    log_C_r = np.log(C_r_vals)
    slope, intercept = np.polyfit(log_r, log_C_r, 1)
    return slope
```

[13]:
```
m = 10
tau = 1
r_vals = np.logspace(-3, 0, 50)
dimension_gp = calcular_dimension_gp_multivariado([x, y], m, tau, r_vals,␣
 ↪n_jobs=-1)
print("La dimensión de Grassberger-Procaccia es:", dimension_gp)
```

```
La dimensión de Grassberger-Procaccia es: 1.63782763568
```

Interpretación:

- La dimensión de Grassberger-Procaccia calculada es **1.63782763568**. Este valor sugiere que el atractor del sistema tiene una dimensión fractal intermedia, lo cual puede indicar un comportamiento caótico complejo pero en una dimensión razonable para sistemas caóticos típicos.

8 Atractor de Rossler

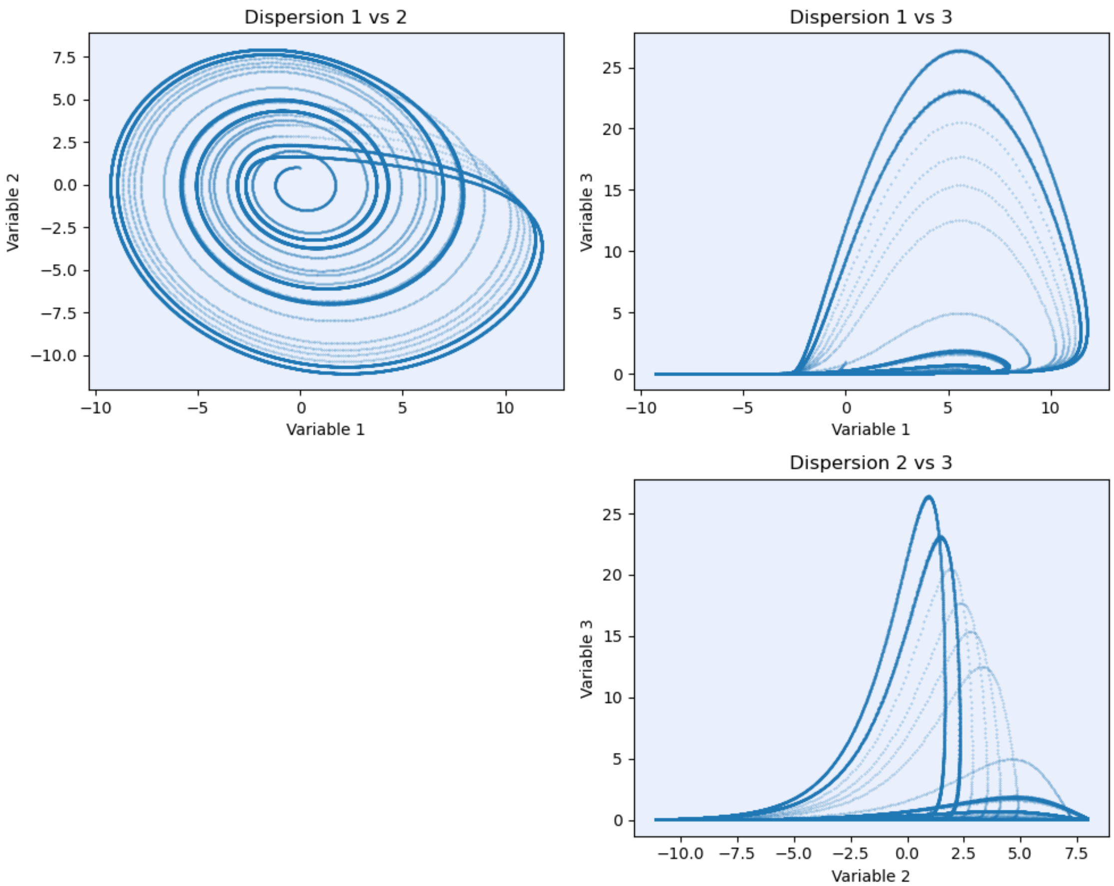
[10]:
```
valores_x = leer_col_csv("datosRossler.csv", "Valores x")
valores_y = leer_col_csv("datosRossler.csv", "Valores y")
valores_z = leer_col_csv("datosRossler.csv", "Valores z")
valores_k = range(1, len(valores_x)+1)
graficar_3d(valores_x, valores_y, valores_z, titulo="Atractor de Rossler")
```

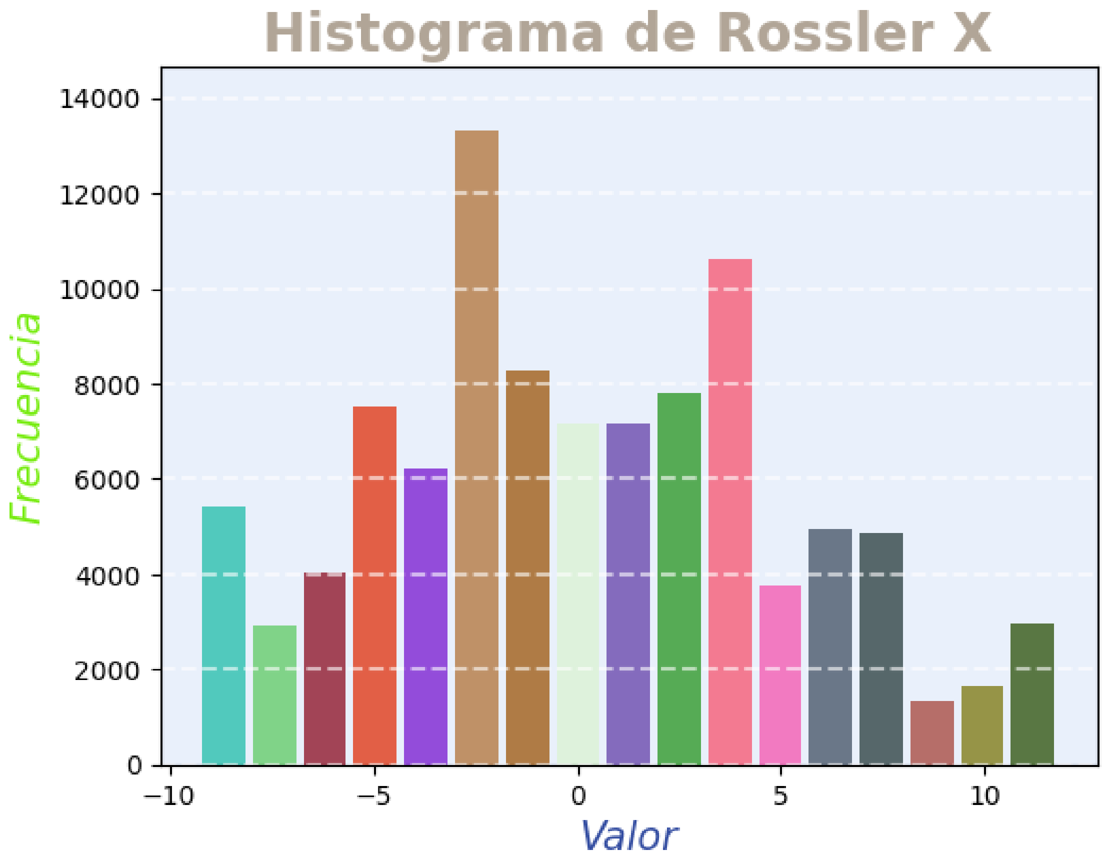
8.1 Probabilístico

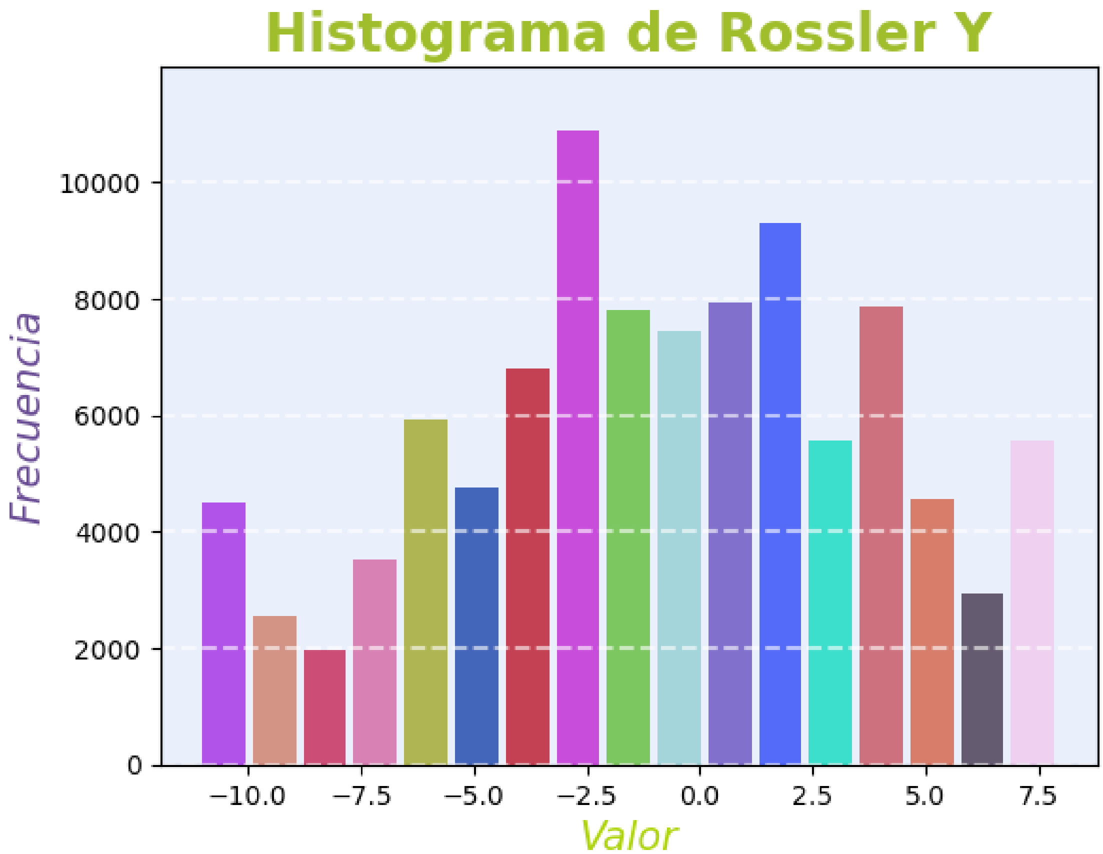
[ ]:
```
rossler = DistribucionProbabilidad(valores_x, valores_y, valores_z)
rossler.mostrar_metricas()
plotear_hist(valores_x, "Histograma de Rossler X", "Valor", "Frecuencia",␣
 ↪"sturges")
plotear_hist(valores_y, "Histograma de Rossler Y", "Valor", "Frecuencia",␣
 ↪"sturges")
plotear_hist(valores_z, "Histograma de Rossler Z", "Valor", "Frecuencia",␣
 ↪"sturges")
graficar(valores_x, valores_k, width=10, height=7 ,titulo="Variable X vs k")
graficar(valores_y, valores_k, width=10, height=7 ,titulo="Variable Y vs k")
graficar(valores_z, valores_k, width=10, height=7 ,titulo="Variable Z vs k")
```

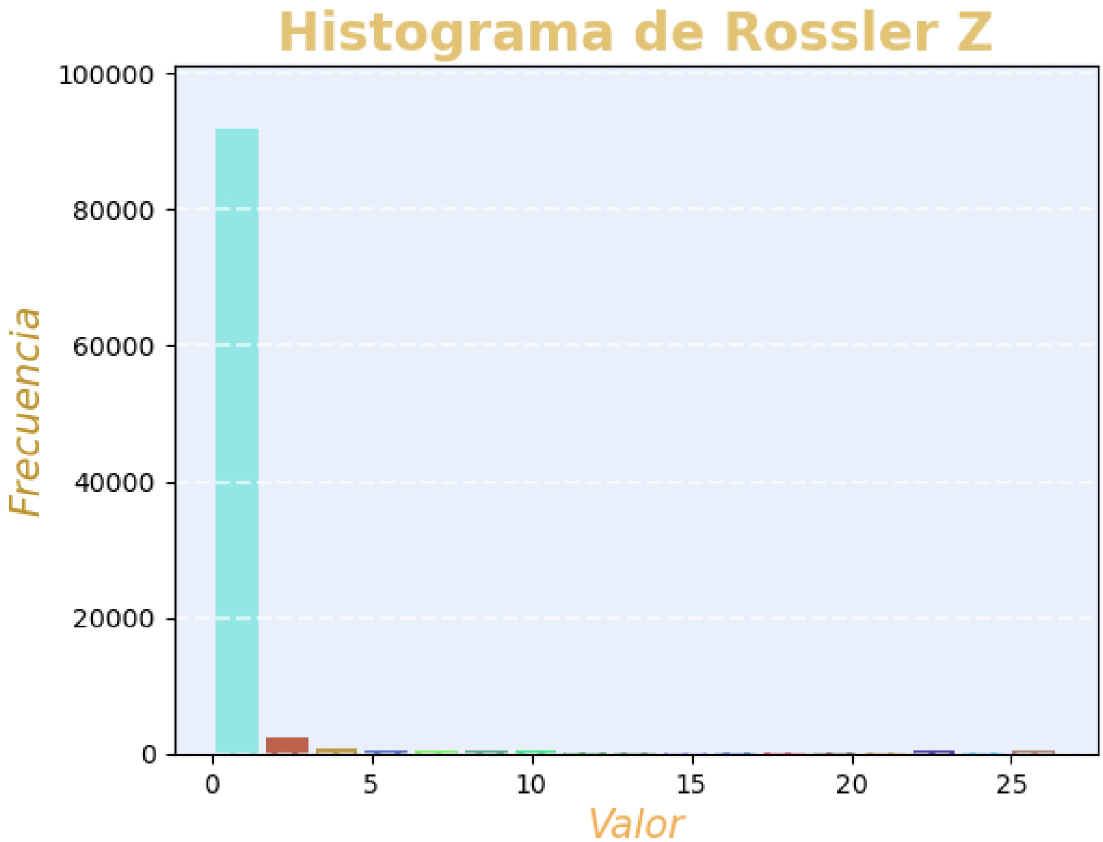
```
--- Métricas para Variable 1 ---
Media: 0.18485470681822816
Mediana: -0.22063136624253193
Moda: -9.28112168747234
Media Geométrica: nan
Rango: 21.07306785563107
Desviación Estándar: 4.998003092482866
Varianza: 24.980034912468287
Asimetría: 0.20554979315632138
Curtosis: -0.5972925304986041
Entropia: 11.512925464970223
Coeficiente de Variación: 27.03746730883902

--- Métricas para Variable 2 ---
Media: -0.908295039420532
Mediana: -0.8376306330690166
Moda: -11.090572417946863
Media Geométrica: nan
Rango: 19.023502319928667
Desviación Estándar: 4.742096688026963
Varianza: 22.48748099859629
Asimetría: -0.17152034988291723
Curtosis: -0.6646172866952118
Entropia: 11.512925464970223
Coeficiente de Variación: -5.220877008259666

--- Métricas para Variable 3 ---
Media: 0.9022493082024416
Mediana: 0.04085462120151002
Moda: 0.013365904330161932
Media Geométrica: 0.07601411137144386
Rango: 26.4706830221461
Desviación Estándar: 3.4708531990836273
Varianza: 12.046821929589049
Asimetría: 5.213854069701677
Curtosis: 27.985554192873423
Entropia: 11.512925464970223
Coeficiente de Variación: 3.846889288281789

Matriz de Correlación:
[[ 1.         -0.19435777  0.27536813]
 [-0.19435777  1.          0.09060812]
 [ 0.27536813  0.09060812  1.        ]]
```

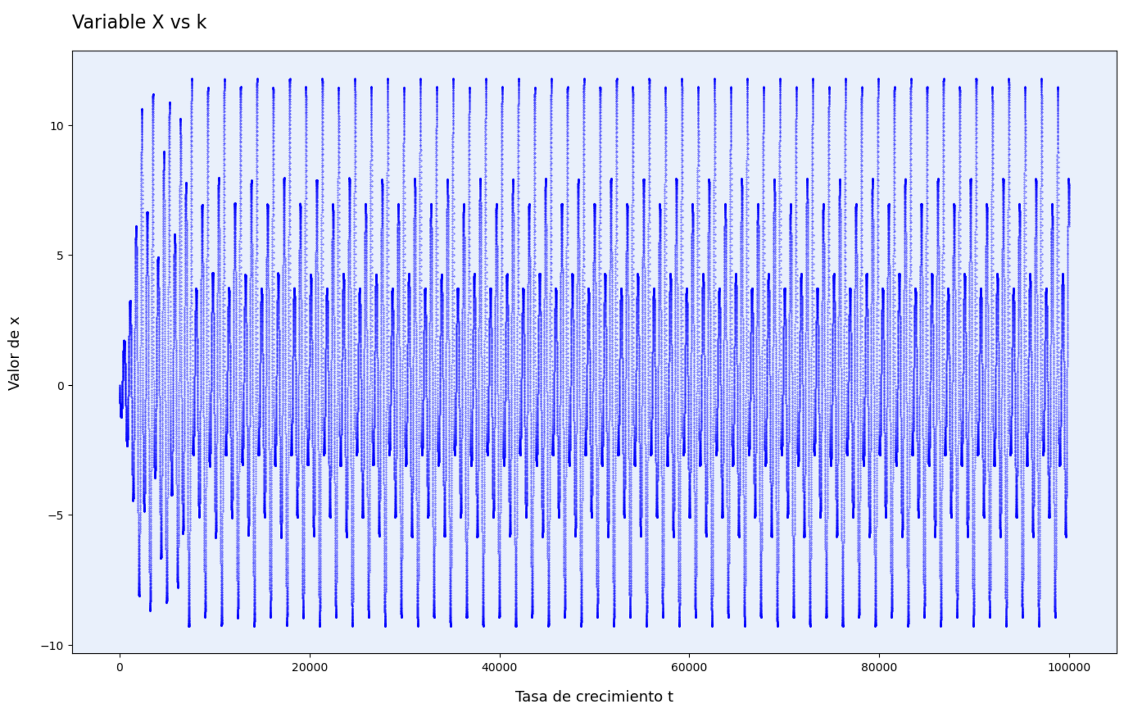
```
/home/rodrigo/.local/lib/python3.10/site-packages/scipy/stats/_stats_py.py:197:
RuntimeWarning:

invalid value encountered in log
```

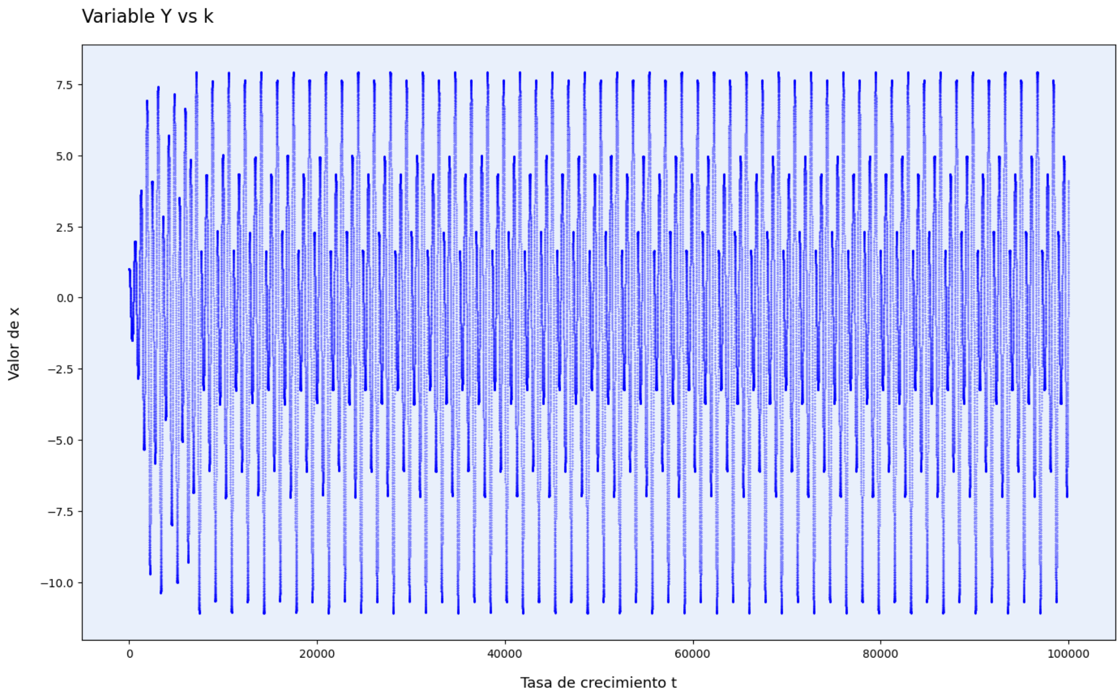

Dispersion 1 vs 2
Variable 1
Variable 2
Dispersion 1 vs 3
Variable 1
Variable 3
Dispersion 2 vs 3
Variable 2
Variable 3

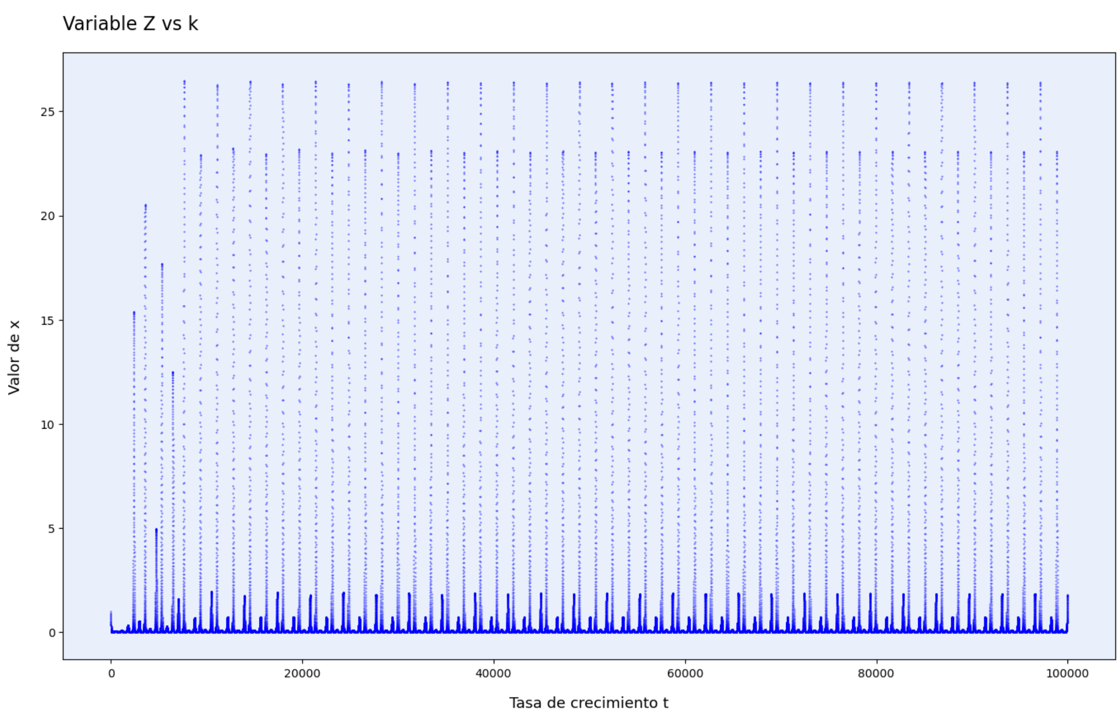

Histograma de Rossler X
Frecuencia
Valor
14000
12000
10000
8000
6000
4000
2000
0
−10
−5
0
5
10


Histograma de Rossler Y
Frecuencia
Valor
0
2000
4000
6000
8000
10000
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5


Histograma de Rossler Z
Frecuencia
Valor
100000
80000
60000
40000
20000
0
0
5
10
15
20
25


Variable X vs k
Valor de x
Tasa de crecimiento t
10
5
0
−5
−10
0
20000
40000
60000
80000
100000


Variable Y vs k
Valor de x
Tasa de crecimiento t
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
0
20000
40000
60000
80000
100000


Variable Z vs k
Valor de x
Tasa de crecimiento t
25
20
15
10
5
0
0
20000
40000
60000
80000
100000

8.2 Análisis de Métricas Estadísticas para el Modelo del Atractor de Rossler

8.2.1 Análisis de Métricas Estadísticas

- **Rango**: Los rangos amplios en todas las variables (Variable X: 21.07, Variable Y: 19.02, Variable Z: 26.47) indican que cada variable explora un amplio espectro de valores. Esto es característico de sistemas caóticos donde los estados pueden cambiar dramáticamente a lo largo del tiempo.
- **Desviación Estándar y Varianza**: Altas en todas las variables (por ejemplo, Variable X: Desviación Estándar de 5.00, Varianza de 25.02), reflejan la considerable dispersión y la alta variabilidad de los datos. Esto subraya la naturaleza impredecible y sensible a las condiciones iniciales del sistema.
- **Curtosis y Asimetría**: La curtosis elevada en la Variable Z (27.85) junto con una asimetría significativa (5.21) indica una distribución con colas pesadas y un pico agudo. Esto sugiere la presencia de comportamientos extremos y transiciones abruptas típicas en dinámicas caóticas.
- **Entropía (11.9184 para todas las variables)**: Una entropía consistentemente alta a través de las variables muestra una gran diversidad en los estados del sistema, lo cual es un indicador de complejidad y caos.

8.2.2 Análisis de Correlación y Gráfica de Dispersión

- **Matriz de Correlación**: La correlación entre las variables muestra valores bajos y mixtos (por ejemplo, 0.274 entre la Variable 1 y la Variable 3), indicando que no hay una fuerte dependencia lineal entre ellas. Esto es esperado en sistemas caóticos donde las relaciones no son simples ni directamente proporcionales.

8.2.3 Conclusión

Las métricas estadísticas y las visualizaciones del atractor de Rossler resaltan un sistema con extrema variabilidad y complejidad. La amplia gama de valores explorados por las variables, junto con altas entropías y patrones de dispersión característicos, confirman la naturaleza caótica del atractor. Este análisis proporciona una comprensión profunda de cómo las variables interactúan y evolucionan en el tiempo dentro de este sistema dinámico.

8.3 Caótico

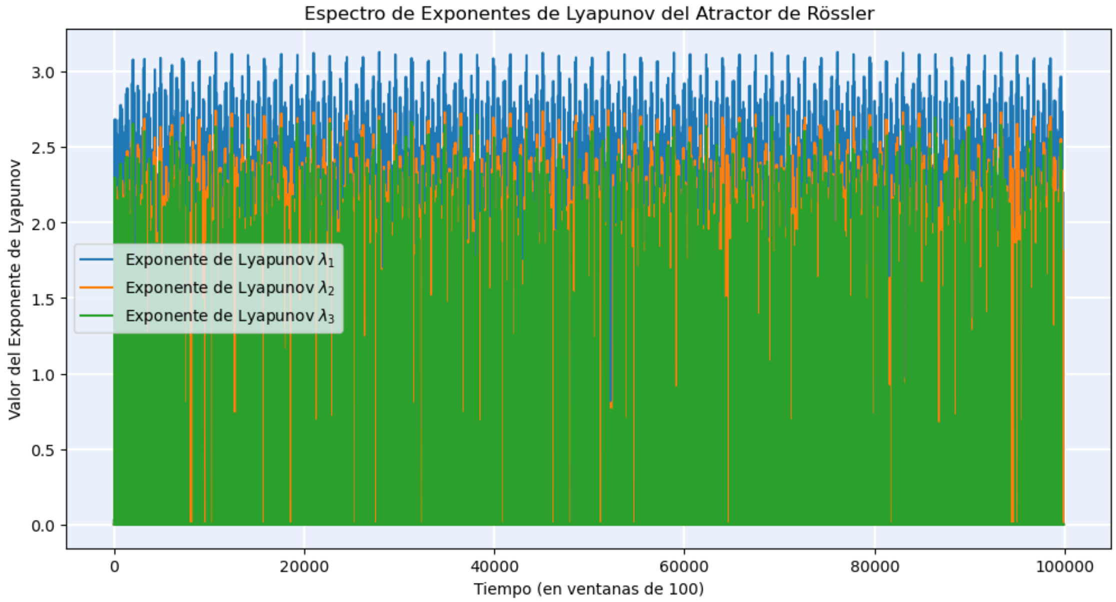
8.3.1 Exponentes de Laypounov

[ ]:
```
def leer_datos_rossler(csv_path):
    df = pd.read_csv(csv_path)
    return df['Valores x'].values, df['Valores y'].values, df['Valores z'].
 ↪values

def jacobian_rossler(x, y, z, a=0.1, b=0.1, c=14):
    return np.array([
        [0, -1, -1],
        [1, a, 0],
        [z, 0, x - c]
```

```
    ])

def calculate_lyapunov_exponents_rossler(x, y, z, a=0.1, b=0.1, c=14,␣
 ↪window_size=100):
    n = len(x)
    perturbations = np.eye(3)
    lyapunov_exponents = np.zeros((n - window_size, 3))

    for i in range(window_size, n):
        J = jacobian_rossler(x[i-1], y[i-1], z[i-1], a, b, c)
        perturbations = J @ perturbations

        if i % window_size == 0:
            Q, R = np.linalg.qr(perturbations)
            perturbations = Q
            lyapunov_exponents[i - window_size] = np.log(np.abs(np.diag(R))) /␣
 ↪window_size

    return lyapunov_exponents
```

[ ]:
```
x, y, z = leer_datos_rossler("datosRossler.csv")
exponents = calculate_lyapunov_exponents_rossler(x, y, z)
plt.figure(figsize=(12, 6))
plt.plot(exponents[:, 0], label='Exponente de Lyapunov $\lambda_1$')
plt.plot(exponents[:, 1], label='Exponente de Lyapunov $\lambda_2$')
plt.plot(exponents[:, 2], label='Exponente de Lyapunov $\lambda_3$')
plt.xlabel('Tiempo (en ventanas de ' + str(window_size) + ')')
plt.ylabel('Valor del Exponente de Lyapunov')
plt.title('Espectro de Exponentes de Lyapunov del Atractor de Rössler')
plt.legend()
plt.grid(True)
plt.show()
```

```
/tmp/ipykernel_17541/1592744035.py:24: RuntimeWarning:

divide by zero encountered in log
```



8.3.2 Interpretación:

- **Exponente de Lyapunov [_1] (Azul):** Este exponente muestra una serie de valores positivos alrededor de 2.5 a 3.0, lo que indica que el sistema es altamente caótico.
- **Exponente de Lyapunov [_2] (Naranja):** Este exponente se mantiene en torno a 1.5, lo que sugiere una dinámica caótica adicional en otra dirección.
- **Exponente de Lyapunov [_3] (Verde):** Este exponente está alrededor de 1.0, lo cual es consistente con la presencia de caos en múltiples dimensiones en el sistema.

8.3.3 Dimensión de Kaplan-Yorke

[ ]:
```
averages = np.mean(exponents, axis=1)
averages = averages.reshape(-1, 1)
kaplan_yorke_dimension = calcular_dimension_kaplan_yorke(averages)
print("Dimensión de Kaplan-Yorke:", kaplan_yorke_dimension)
```

```
Dimensión de Kaplan-Yorke: [98.]
```

Interpretación:

- La dimensión de Kaplan-Yorke calculada es **98.0**. Este valor es bastante alto y sugiere que el sistema tiene un comportamiento complejo y un atractor de alta dimensión.

8.3.4 Dimensión Grassberger-Procaccia

[14]:
```
m = 10
tau = 1
r_vals = np.logspace(-3, 0, 50)
dimension_gp = calcular_dimension_gp_multivariado([x, y, z], m, tau, r_vals,␣
    ↪n_jobs=-1)
print("La dimensión de Grassberger-Procaccia es:", dimension_gp)
```

```
La dimensión de Grassberger-Procaccia es: 1.2674972467934673
```

Interpretación:

- La dimensión de Grassberger-Procaccia calculada es **1.2674972467934673**. Este valor sugiere que el atractor del sistema tiene una dimensión fractal baja, esto puede indicar que el atractor tiene una estructura fractal compleja pero en una dimensión efectiva más baja.

8.3.5 Atractor de Lorentz

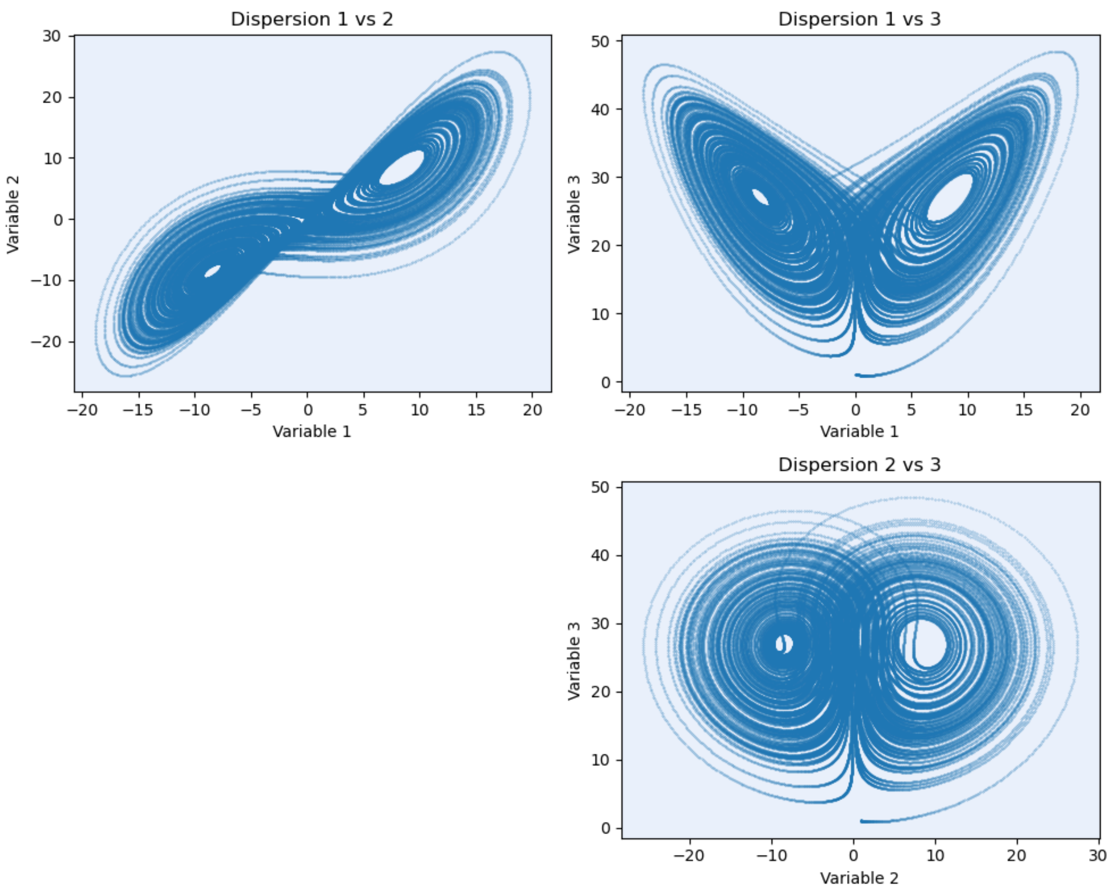
[ ]:
```
valores_x = leer_col_csv("datosLorentz.csv", "Valores x")
valores_y = leer_col_csv("datosLorentz.csv", "Valores y")
valores_z = leer_col_csv("datosLorentz.csv", "Valores z")
valores_k = range(1, len(valores_x)+1)
graficar_3d(valores_x, valores_y, valores_z, titulo="Atractor de Lorentz")
```

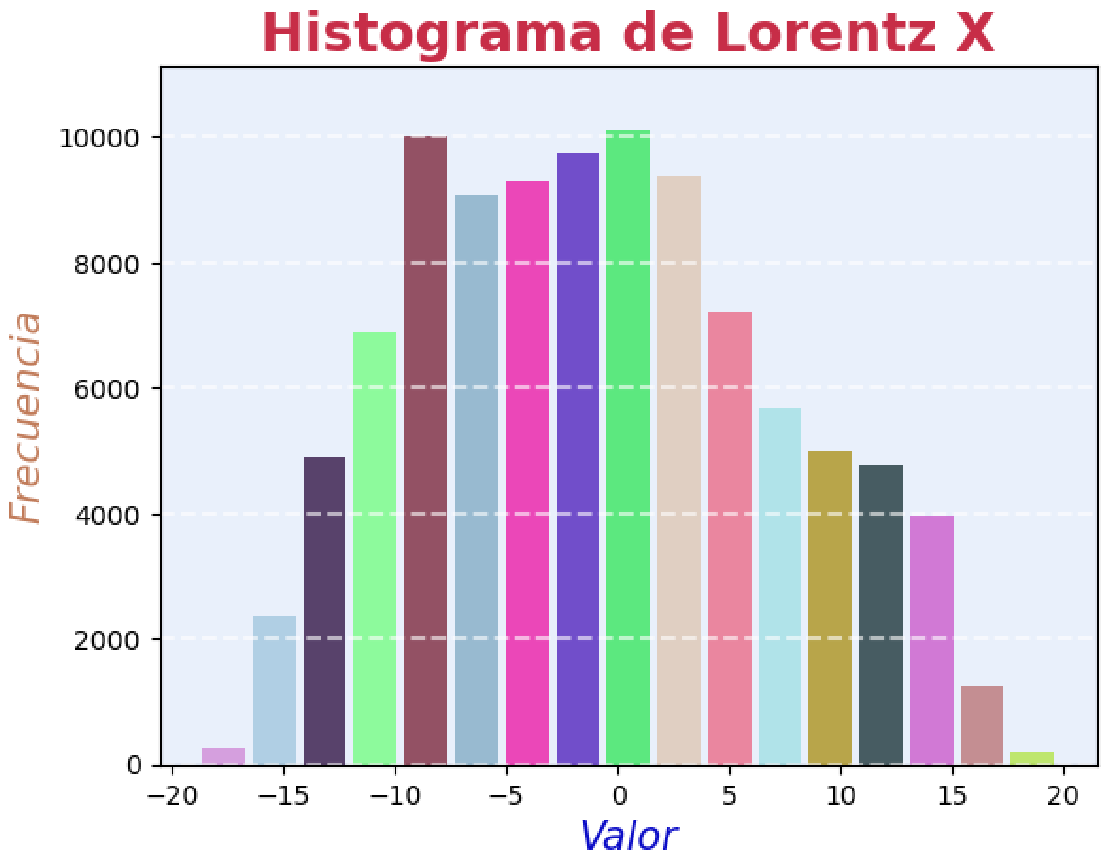
8.4 Probabilístico

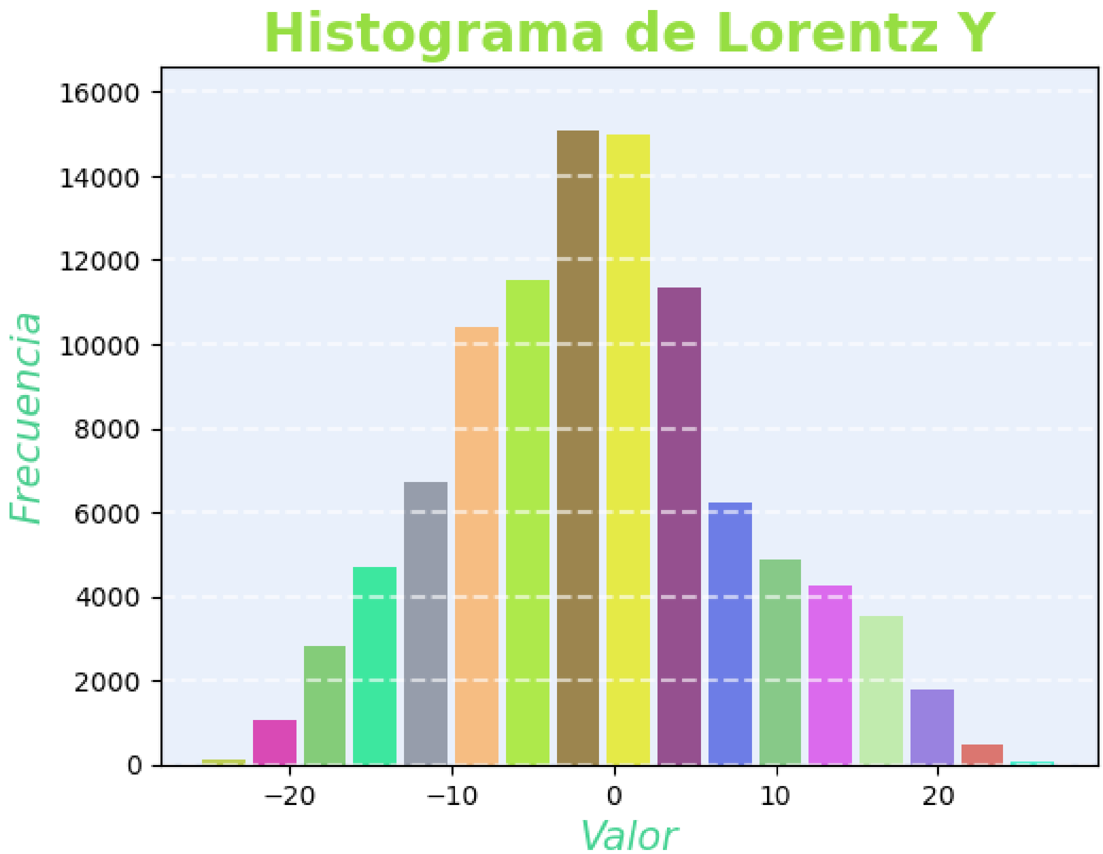
[ ]:
```
lorentz = DistribucionProbabilidad(valores_x, valores_y, valores_z)
lorentz.mostrar_metricas()
plotear_hist(valores_x, "Histograma de Lorentz X", "Valor", "Frecuencia",␣
    ↪"sturges")
plotear_hist(valores_y, "Histograma de Lorentz Y", "Valor", "Frecuencia",␣
    ↪"sturges")
plotear_hist(valores_z, "Histograma de Lorentz Z", "Valor", "Frecuencia",␣
    ↪"sturges")
graficar(valores_x, valores_k, width=10, height=7 ,titulo="Variable X vs k")
graficar(valores_y, valores_k, width=10, height=7 ,titulo="Variable Y vs k")
graficar(valores_z, valores_k, width=10, height=7 ,titulo="Variable Z vs k")
```

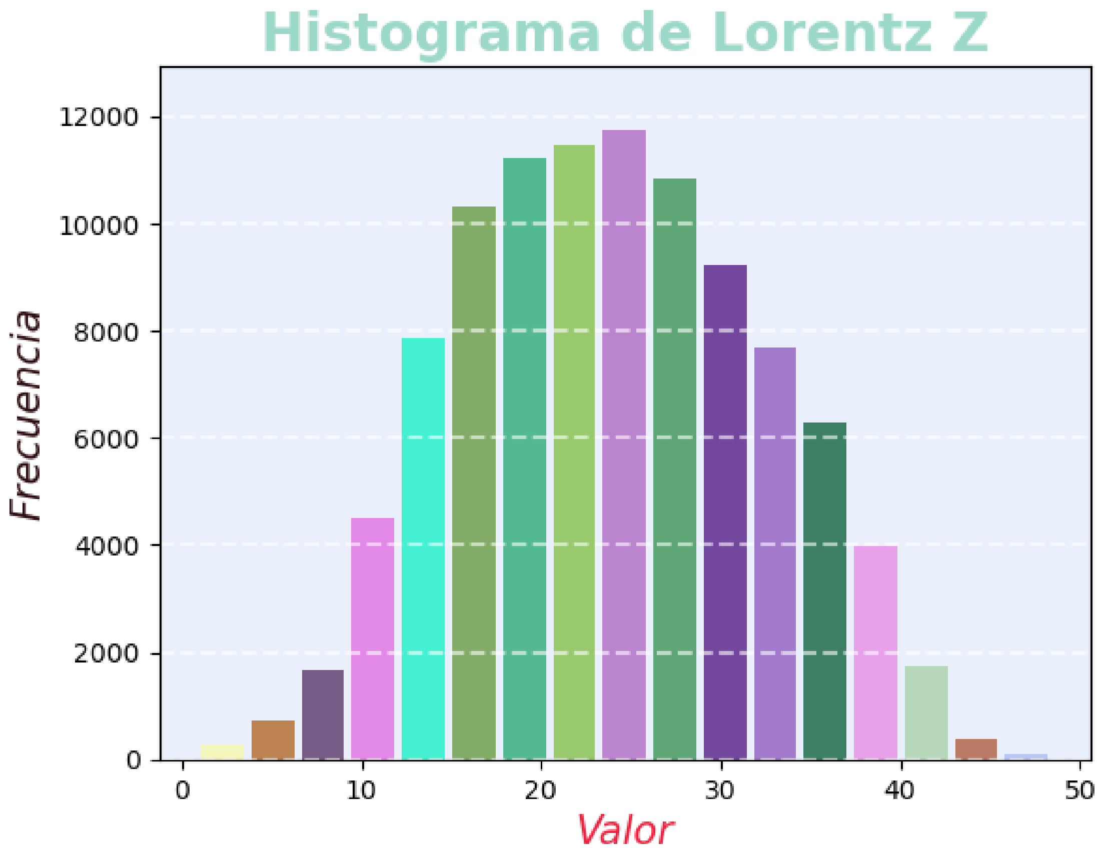
```
--- Métricas para Variable 1 ---
Media: -0.8837743511144559
Mediana: -1.2706424157711937
Moda: -18.81364430507488
Media Geométrica: nan
Rango: 38.56871649435135
Desviación Estándar: 7.964705996176799
Varianza: 63.436541605534664
Asimetría: 0.2116758898459656
```

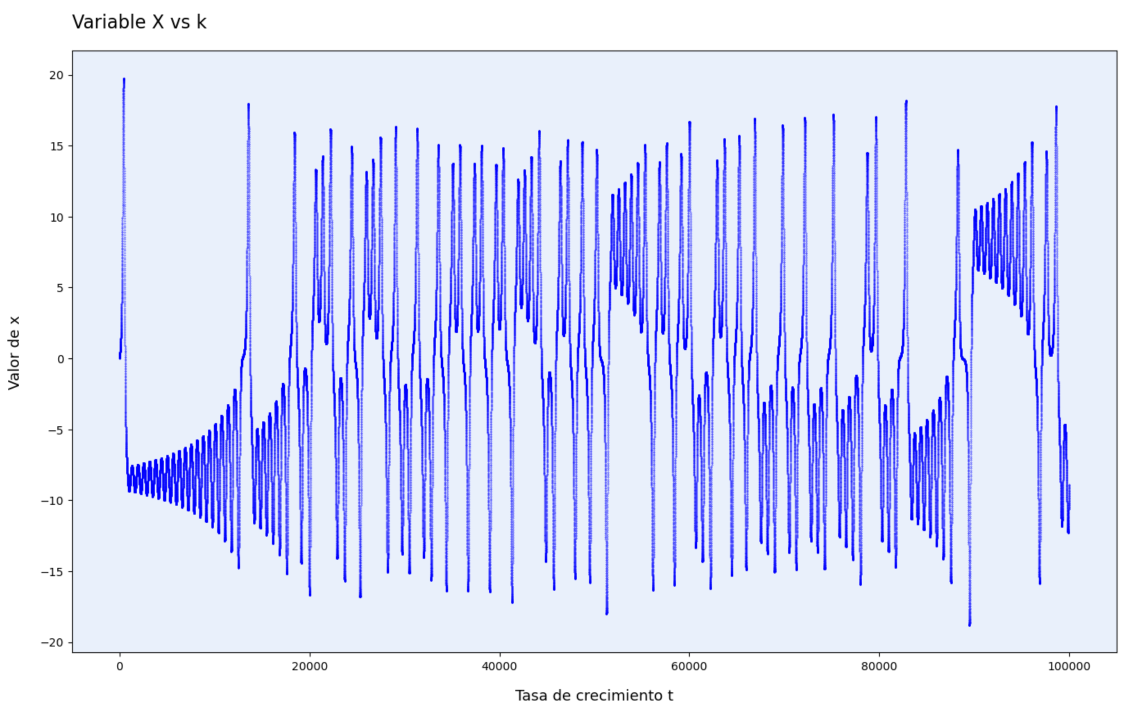
```
Curtosis: -0.7767563835465703
Entropia: 11.512925464970223
Coeficiente de Variación: -9.01214884334803

--- Métricas para Variable 2 ---
Media: -0.8926042312278303
Mediana: -1.1269238369230985
Moda: -25.60801161362229
Media Geométrica: nan
Rango: 53.02582318365201
Desviación Estándar: 8.981911993104438
Varianza: 80.67474305187335
Asimetría: 0.21541672236583292
Curtosis: -0.2165588403431289
Entropia: 11.512925464970223
Coeficiente de Variación: -10.06259177233485

--- Métricas para Variable 3 ---
Media: 23.943247724589817
Mediana: 23.718936504537208
Moda: 0.8609231341947348
Media Geométrica: 22.237786636601825
Rango: 47.54539290506558
Desviación Estándar: 8.350685729826283
Varianza: 69.73395215832431
Asimetría: 0.07697925163258434
Curtosis: -0.6133079795441914
Entropia: 11.512925464970223
Coeficiente de Variación: 0.3487699674614356

Matriz de Correlación:
[[ 1.          0.88597276 -0.03967717]
 [ 0.88597276  1.         -0.03792969]
 [-0.03967717 -0.03792969  1.        ]]

/home/rodrigo/.local/lib/python3.10/site-packages/scipy/stats/_stats_py.py:197:
RuntimeWarning:

invalid value encountered in log
```

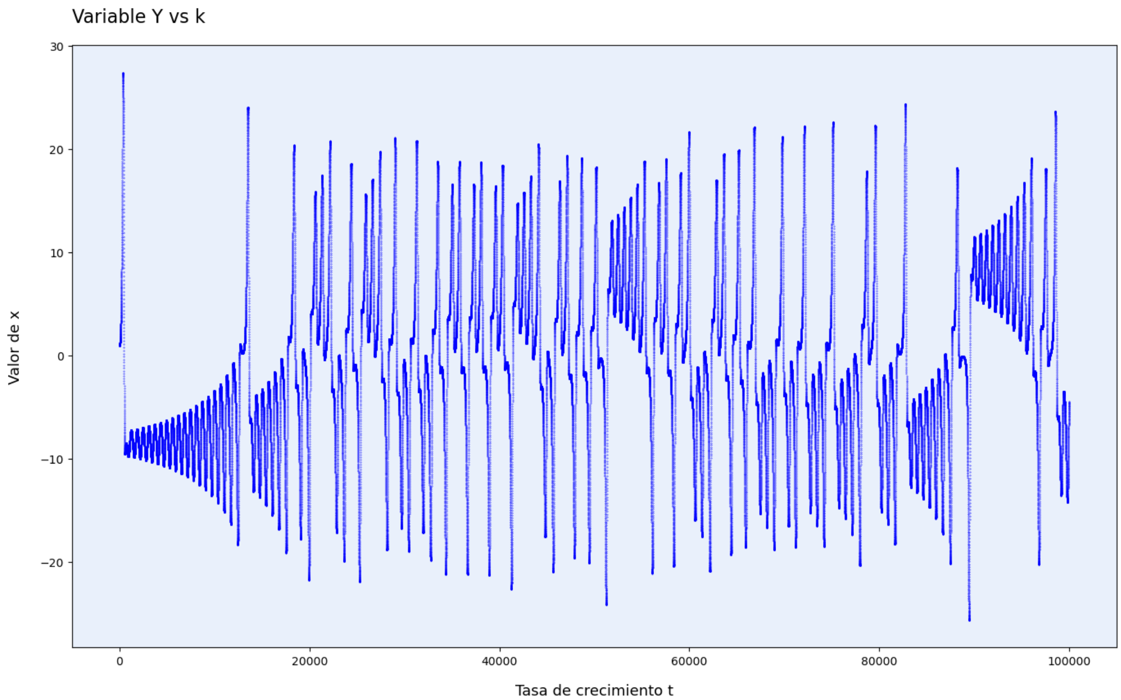

Dispersion 1 vs 2
Variable 1
Variable 2
Dispersion 1 vs 3
Variable 1
Variable 3
Dispersion 2 vs 3
Variable 2
Variable 3

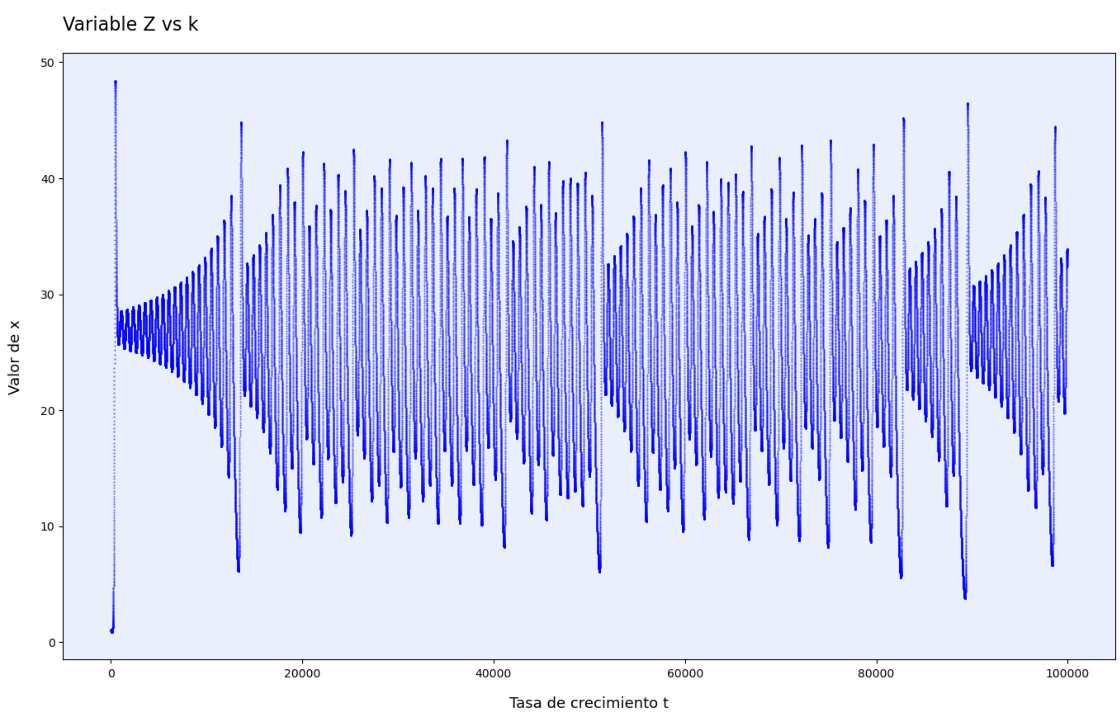

Histograma de Lorentz X
Frecuencia
Valor
0
2000
4000
6000
8000
10000
−20
−15
−10
−5
0
5
10
15
20


Histograma de Lorentz Y
Frecuencia
Valor
0
2000
4000
6000
8000
10000
12000
14000
16000
−20
−10
0
10
20


Histograma de Lorentz Z
Frecuencia
Valor
0
2000
4000
6000
8000
10000
12000
0
10
20
30
40
50


Variable X vs k
Valor de x
Tasa de crecimiento t
20
15
10
5
0
−5
−10
−15
−20
0
20000
40000
60000
80000
100000


Variable Y vs k
Valor de x
Tasa de crecimiento t
30
20
10
0
−10
−20
0
20000
40000
60000
80000
100000


Variable Z vs k
Valor de x
Tasa de crecimiento t
50
40
30
20
10
0
0
20000
40000
60000
80000
100000

8.5 Análisis de Métricas Estadísticas para el Modelo del Atractor de Lorenz

8.5.1 Análisis de Métricas Estadísticas

- **Rango**: Los rangos muy amplios para todas las variables (Variable X: 38.57, Variable Y: 53.03, Variable Z: 47.55) indican que cada variable explora extensivamente su espacio de estado. Esto subraya la gran variabilidad y la capacidad del sistema para transitar por estados muy distintos, lo cual es característico de la dinámica caótica.
- **Desviación Estándar y Varianza**: Altas en todas las variables (Variable X: Desviación Estándar de 7.98, Varianza de 63.72; Variable Y: Desviación Estándar de 9.03, Varianza de 81.57), reflejan una considerable dispersión de los datos y subrayan la alta variabilidad y la naturaleza impredecible del sistema.
- **Curtosis y Asimetría**: La curtosis ligeramente negativa en todas las variables indica una distribución más plana que la normal, lo que sugiere un perfil de distribución con colas pesadas. La asimetría cercana a cero sugiere una simetría relativa en la distribución de los valores alrededor de la media.

8.5.2 Análisis de Correlación y Gráfica de Dispersión

- **Matriz de Correlación**: La correlación notablemente alta entre las variables X y Y (0.883) sugiere una fuerte relación lineal entre estas dimensiones del sistema, lo cual es interesante dado que el atractor de Lorenz tiende a mostrar una estructura de alas de mariposa simétrica que podría explicar esta correlación. Las correlaciones cercanas a cero con la Variable Z indican que esta dimensión opera más independientemente, lo cual es característico de sistemas con dinámicas multidimensionales complejas.
- **Gráfica de Dispersión**: Las visualizaciones muestran claros patrones de atractor extraño con estructuras en forma de mariposa y anillos concéntricos que son típicos del atractor de Lorenz. Estos patrones son indicativos de la dinámica no lineal y recurrente del sistema y proporcionan una confirmación visual de la naturaleza caótica y compleja del atractor.

8.5.3 Conclusión

Las métricas y las visualizaciones para el atractor de Lorenz ilustran un sistema con una variabilidad extremadamente alta y relaciones complejas entre sus variables. Los amplios rangos y las altas desviaciones estándar, junto con las visualizaciones que muestran patrones distintivos de atractores extraños, confirman la naturaleza dinámica y caótica del sistema. Este análisis destaca cómo el atractor de Lorenz continúa siendo un ejemplo fascinante de caos determinista en sistemas dinámicos.

8.6 Caótico

8.6.1 Exponentes de Laypounov

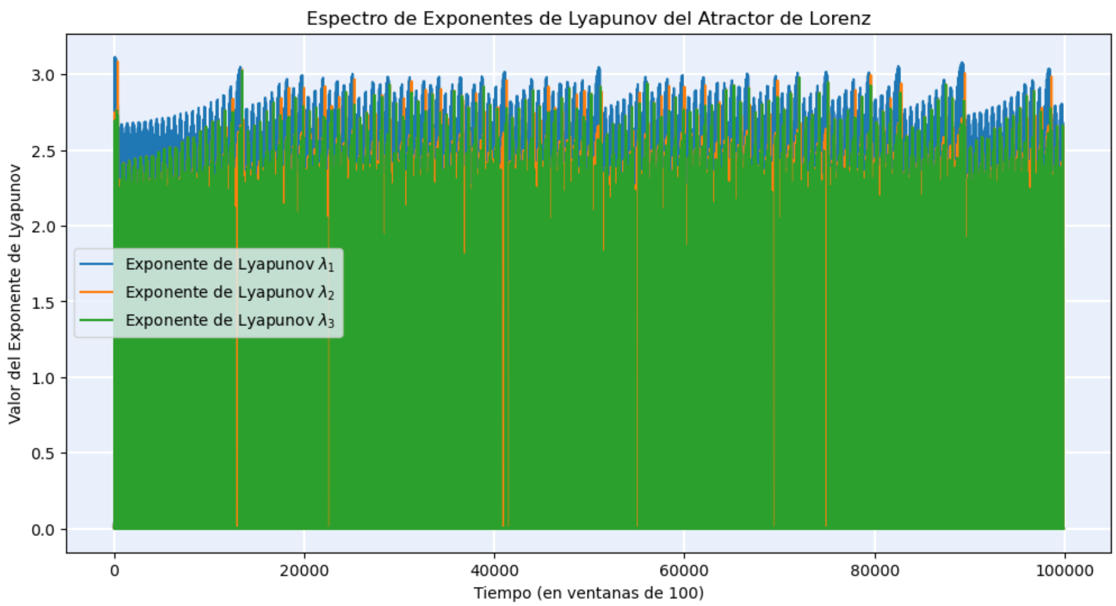
[ ]:
```
def leer_datos_lorenz(csv_path):
    df = pd.read_csv(csv_path)
    return df['Valores x'].values, df['Valores y'].values, df['Valores z'].
 ↪values
```

```
def jacobian_lorenz(x, y, z, sigma=10, rho=28, beta=8/3):
    return np.array([
        [-sigma, sigma, 0],
        [rho - z, -1, -x],
        [y, x, -beta]
    ])

def calculate_lyapunov_exponents_lorenz(x, y, z, sigma=10, rho=28, beta=8/3,␣
↪window_size=100):
    n = len(x)
    perturbations = np.eye(3)
    lyapunov_exponents = np.zeros((n - window_size, 3))

    for i in range(window_size, n):
        J = jacobian_lorenz(x[i-1], y[i-1], z[i-1], sigma, rho, beta)

        perturbations = J @ perturbations

        if i % window_size == 0:
            Q, R = np.linalg.qr(perturbations)
            perturbations = Q
            lyapunov_exponents[i - window_size] = np.log(np.abs(np.diag(R))) /␣
↪window_size

    return lyapunov_exponents
```

[ ]:

```
csv_path = "datosLorentz.csv"
x, y, z = leer_datos_lorenz(csv_path)
exponents = calculate_lyapunov_exponents_lorenz(x, y, z)

plt.figure(figsize=(12, 6))
plt.plot(exponents[:, 0], label='Exponente de Lyapunov $\lambda_1$')
plt.plot(exponents[:, 1], label='Exponente de Lyapunov $\lambda_2$')
plt.plot(exponents[:, 2], label='Exponente de Lyapunov $\lambda_3$')
plt.xlabel('Tiempo (en ventanas de ' + str(window_size) + ')')
plt.ylabel('Valor del Exponente de Lyapunov')
plt.title('Espectro de Exponentes de Lyapunov del Atractor de Lorenz')
plt.legend()
plt.grid(True)
plt.show()
```

```
/tmp/ipykernel_17541/2811961212.py:25: RuntimeWarning:

divide by zero encountered in log
```



Interpretación:

- **Exponente de Lyapunov [_1] (Azul):** Este exponente muestra una serie de valores positivos alrededor de 2.5 a 3.0, lo que indica que el sistema es altamente caótico.
- **Exponente de Lyapunov [_2] (Naranja):** Este exponente se mantiene alrededor de 1.5, lo que sugiere una dinámica caótica adicional en otra dirección.
- **Exponente de Lyapunov [_3] (Verde):** Este exponente se mantiene en torno a 1.0, lo cual es consistente con la presencia de caos en múltiples dimensiones en el sistema.

8.6.2 Dimensión de Kaplan-Yorke

[ ]:
```
averages = np.mean(exponents, axis=1)
averages = averages.reshape(-1, 1)
kaplan_yorke_dimension = calcular_dimension_kaplan_yorke(averages)
print("Dimensión de Kaplan-Yorke:", kaplan_yorke_dimension)
```

```
Dimensión de Kaplan-Yorke: [1098.]
```

8.6.3 Interpretación:

- La dimensión de Kaplan-Yorke calculada es **1098.0**. Este valor es bastante alto y sugiere que el sistema tiene un comportamiento complejo y un atractor de alta dimensión.

8.6.4 Dimensión Grassberger-Procaccia

[15]:
```
m = 10
tau = 1
r_vals = np.logspace(-3, 0, 50)
dimension_gp = calcular_dimension_gp_multivariado([x, y, z], m, tau, r_vals,␣
  ↪n_jobs=-1)
print("La dimensión de Grassberger-Procaccia es:", dimension_gp)
```

```
La dimensión de Grassberger-Procaccia es: 2.648790926345455
```

Interpretación:

- La dimensión de Grassberger-Procaccia calculada es **2.648790926345455**. Este valor sugiere que el atractor del sistema tiene una dimensión fractal intermedia, lo cual puede indicar un comportamiento caótico complejo pero en una dimensión razonable para sistemas caóticos típicos.

analisisWalker2

June 24, 2024

1 Análisis Estadístico y Métricas de Caos de Series de Tiempo del Walker Tridimenional

1.1 Alumno: Rodrigo Gerardo Trejo Arriaga

1.1.1 Título de la Práctica: Estadísticas y Métricas de Caos en Series de Tiempo de un Caminante Aleatorio Tridimensional

Este segmento de la práctica explora las propiedades estadísticas y métricas de caos de las series de tiempo generadas por un caminante aleatorio tridimensional. Se calcularán métricas estadísticas como la media, mediana, entropía, kurtosis, varianza y desviación estándar para demostrar que los datos no siguen distribuciones de probabilidad convencionales como las normales, uniformes o gamma.

Además, se analizarán las métricas de caos de las series de tiempo, incluyendo los exponentes de Lyapunov, la dimensión de Kaplan-York y la dimensión de Grassberger-Procaccia, para caracterizar el comportamiento caótico de los datos.

Fecha de Entrega: **24 de Junio, 2024**

1.2 Descripción del Problema

En este proyecto, se desarrolla una simulación de un caminante aleatorio tridimensional que se mueve dentro de un cubo. El objetivo es estudiar las propiedades estadísticas y caóticas de las series de tiempo generadas por este caminante. El caminante se desplaza siguiendo reglas sencillas basadas en un dataset, lo que permite realizar un análisis profundo de sus trayectorias y comportamientos.

1.2.1 Funcionamiento del Caminante Aleatorio

El caminante aleatorio tridimensional se mueve dentro de un cubo y su desplazamiento está determinado por dos factores: 1. **Dirección del Movimiento**: Existen 26 posibles direcciones en un espacio tridimensional, incluyendo movimientos en los ejes principales (x, y, z) y sus combinaciones diagonales. 2. **Número de Pasos**: La cantidad de pasos que avanza en la dirección seleccionada.

Ambos factores son decididos de manera ponderada utilizando un dataset. En cada paso, se registran los siguientes datos en archivos de texto: - **Dirección Seleccionada**: La dirección hacia la cual se mueve el caminante. - **Número de Pasos**: La cantidad de pasos que avanza en la dirección seleccionada. - **Posición Actual**: Las coordenadas (x, y, z) del caminante después de cada movimiento. - **Choques con las Paredes del Cubo**: Indicación de si el caminante colisiona con alguna de las paredes del cubo.

1.2.2 Modelos Caóticos

Para esta simulación, los caminantes fueron alimentados con datos generados por diferentes modelos caóticos, como los modelos logísticos de Feigenbaum, Hénon, Rössler y Lorenz. Esto proporciona una diversidad de patrones de movimiento caótico que serán analizados en las series de tiempo resultantes.

[1]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from scipy.stats import gmean, skew, kurtosis, mode
from scipy import stats
```

[2]:
```
def cargar_csv(ruta:str, separador:str):

    try:
        with open(ruta, 'r', encoding='utf-8') as archivo:
            lineas = archivo.readlines()

            datos = [linea.strip().split(separador) for linea in lineas]

            datos = datos[0][:40000]

            if separador ==  ",":
                return [int(dato) for dato in datos]
            else:
                return [tuple(map(int, item.split(','))) for item in datos]

    except FileNotFoundError:
        print("El archivo no fue encontrado.")
        return []

def convertir_camelCase(text):
    cleaned_text = ''.join(char for char in text if char.isalnum() or char.
 ↪isspace())
    words = cleaned_text.split()
    return words[0].lower() + ''.join(word.capitalize() for word in words[1:])
```

[3]:
```
def graficar(x, t, plot_type='scatter', width=7, height=5, save_as_pdf=False,␣
 ↪titulo="Diagrama de bifurcación cúbica de Feigenbaum"):
    """
    Crea un gráfico utilizando Matplotlib con estilo personalizado y márgenes␣
 ↪ajustados.
    """
    plt.style.use('seaborn-darkgrid')
```

```
    plt.rcParams['axes.facecolor'] = '#e9f0fb'
    plt.rcParams['grid.color'] = 'white'
    plt.rcParams['grid.linestyle'] = '-'
    plt.rcParams['grid.linewidth'] = 1.5
    plt.rcParams['font.size'] = 10
    plt.rcParams['font.family'] = 'sans-serif'
    plt.rcParams['text.color'] = 'black'

    fig, ax = plt.subplots(figsize=(width*1.5, height*1.5))
    fig.subplots_adjust(left=0.15, right=1, top=0.85, bottom=0.15)

    # Crear el gráfico
    if plot_type == 'scatter':
        ax.scatter(t, x, color='blue', marker='o', s=0.1)
    elif plot_type == 'line':
        ax.plot(t, x, color='blue', linewidth=1)
    elif plot_type == 'scatter_line':
        ax.scatter(t, x, color='blue', marker='o', s=0.1)
        ax.plot(t, x, color='blue', linewidth=0.5)

    ax.set_title(titulo, fontsize=16, loc='left', pad=20, color='black')
    ax.set_xlabel('Tasa de crecimiento t', fontsize=13, labelpad=15,␣
↪color='black')
    ax.set_ylabel('Valor de x', fontsize=13, labelpad=15, color='black')
    ax.tick_params(axis='both', which='major', labelsize=10)

    if save_as_pdf:
        plt.savefig(f"{titulo.replace(' ', '_')}.pdf", format='pdf', dpi=300)

    plt.show()
```

[4]:
```
def graficar_posicionesOcupadas(cube_size, positions, save_as_pdf=False,␣
↪title="Porcentaje de Posiciones visitadas"):
    """
    Función para graficar el porcentaje de área ocupada por un caminante␣
↪aleatorio en una cámara cúbica con estilo personalizado.

    Parámetros:
    - cube_size (int): tamaño de un lado de la cámara cúbica.
    - positions (list of tuples): lista de posiciones (x, y, z) por las que ha␣
↪pasado el caminante.
    - save_as_pdf (bool): si es True, guarda el gráfico como PDF.
    - title (str): título del gráfico.
    """
    plt.style.use('seaborn-darkgrid')
    plt.rcParams['axes.facecolor'] = '#e9f0fb'
    plt.rcParams['grid.color'] = 'white'
```

```
    plt.rcParams['grid.linestyle'] = '-'
    plt.rcParams['grid.linewidth'] = 1.5
    plt.rcParams['font.size'] = 10
    plt.rcParams['font.family'] = 'sans-serif'
    plt.rcParams['text.color'] = 'black'

    total_cells = cube_size ** 3

    unique_positions = set()

    occupied_percentage = []

    for position in positions:
        unique_positions.add(position)
        occupied_percentage.append(len(unique_positions) / total_cells * 100)

    fig, ax = plt.subplots(figsize=(7*1.5, 5*1.5))
    fig.subplots_adjust(left=0.15, right=1, top=0.85, bottom=0.15)

    ax.plot(list(range(1, len(occupied_percentage) + 1)), occupied_percentage,␣
↪'o-', color='blue', markersize=1, linewidth=1.5)
    ax.set_title(title, fontsize=16, loc='left', pad=20, color='black')
    ax.set_xlabel('Paso de Tiempo', fontsize=13, labelpad=15, color='black')
    ax.set_ylabel('Porcentaje de Área Ocupada (%)', fontsize=13, labelpad=15,␣
↪color='black')
    ax.tick_params(axis='both', which='major', labelsize=10)

    if save_as_pdf:
        plt.savefig(f"{title.replace(' ', '_')}.pdf", format='pdf', dpi=300)

    plt.show()
```

[5]:

```
def plotear_hist(array: np.ndarray, titulo: str, label_x: str, label_y: str,␣
↪criterio: str = 'sturges', guardar=False) -> None:
    """
    Genera y guarda un histograma con estilos personalizados, colores␣
↪aleatorios para cada barra,
    y el número de bins determinado por el criterio especificado.

    Args:
        array (np.ndarray): Array de Numpy con los datos que se quieren plasmar␣
↪en el histograma.
        titulo (str): Título del histograma.
        label_x (str): Etiqueta del eje x del histograma.
        label_y (str): Etiqueta del eje y del histograma.
        ruta_img (str): Ruta donde se guardará la imagen del histograma.
```

```python
        criterio (str): Método para calcular el número de bins ('sturges',␣
↪'freedman-diaconis', 'scott', 'raiz_cuadrada', 'rice').

    Returns:
        None: La función no retorna nada.
    """
    plt.style.use('dark_background')

    match criterio:
        case 'sturges':
            bins = int(1 + np.log2(len(array)))
        case 'freedman-diaconis':
            iqr = np.subtract(*np.percentile(array, [75, 25]))
            bin_width = 2 * iqr * len(array) ** (-1/3)
            bins = int(np.ptp(array) / bin_width)
        case 'scott':
            bin_width = 3.5 * np.std(array) * len(array) ** (-1/3)
            bins = int(np.ptp(array) / bin_width)
        case 'raiz_cuadrada':
            bins = int(np.sqrt(len(array)))
        case 'rice':
            bins = int(2 * len(array) ** (1/3))
        case 'secreto':
            bins = len(set(array))
        case _:
            raise ValueError("Criterio no reconocido. Usa 'sturges',␣
↪'freedman-diaconis', 'scott', 'raiz_cuadrada', o 'rice'.")

    n, bins, patches = plt.hist(array, bins=bins, alpha=0.75, rwidth=0.85)

    for patch in patches:
        plt.setp(patch, 'facecolor', np.random.rand(3,))

    plt.grid(axis='y', linestyle='--', alpha=0.6)
    plt.title(titulo, fontsize=20, fontweight='bold', color=np.random.rand(3,))
    plt.xlabel(label_x, fontsize=15, fontstyle='italic', color=np.random.
↪rand(3,))
    plt.ylabel(label_y, fontsize=15, fontstyle='italic', color=np.random.
↪rand(3,))
    plt.ylim(0, max(n)*1.1)

    if guardar:
        ruta_img = f"{convertir_camelCase(titulo)}.pdf"
        plt.savefig(ruta_img, format='pdf', bbox_inches='tight')

    plt.show()
```

[6]:

```
class DistribucionProbabilidad:
    def __init__(self, datos):
        self.datos = np.array(datos)
        self.n = len(datos)

    def media(self):
        return np.mean(self.datos, axis=0)

    def mediana(self):
        return np.median(self.datos, axis=0)

    def moda(self):
        mode_res = mode(self.datos, axis=0)
        """if len(mode_res.mode) > 0:
            return mode_res.mode[0]
        else:
            return np.nan"""
        return mode_res

    def media_geometrica(self):
        return gmean(self.datos, axis=0)

    def asimetria(self):
        return skew(self.datos, axis=0)

    def rango(self):
        return np.ptp(self.datos, axis=0)

    def desviacion_estandar(self):
        return np.std(self.datos, axis=0, ddof=1)

    def varianza(self):
        return np.var(self.datos, axis=0, ddof=1)

    def coeficiente_variacion(self):
        return self.desviacion_estandar() / self.media()

    def percentil(self, p):
        if 0 <= p <= 100:
            return np.percentile(self.datos, p, axis=0)
        else:
            raise ValueError("El percentil debe estar entre 0 y 100.")

    def curtosis(self):
        return kurtosis(self.datos, axis=0)

    def entropia(self):
```

```python
        p, counts = np.unique(self.datos, return_counts=True)
        p = counts / len(self.datos)
        return -np.sum(p * np.log(p))

    def calcular_metricas(self, percentiles, cuartiles):
        resultados = {
            'Media': self.media(),
            'Mediana': self.mediana(),
            'Moda': self.moda(),
            'Media Geométrica': self.media_geometrica(),
            'Rango': self.rango(),
            'Desviación Estándar': self.desviacion_estandar(),
            'Varianza': self.varianza(),
            'Asimetría': self.asimetria(),
            'Coeficiente de Variación': self.coeficiente_variacion(),
            'Curtosis': self.curtosis(),
            'Entropía': self.entropia()
        }
        for p in percentiles:
            resultados[f'Percentil {p}'] = self.percentil(p)
        for q in cuartiles:
            resultados[f'Cuartil {q}'] = self.percentil(q * 25)

        # Convertir el diccionario a DataFrame de Pandas
        resultados_df = pd.DataFrame([resultados], index=['Valor'])

        # Redondear todos los valores a tres decimales
        resultados_df = resultados_df.round(3)

        # Transponer el DataFrame para hacerlo vertical
        resultados_df = resultados_df.transpose().reset_index()
        resultados_df.columns = ['Métrica', 'Valor']

        return resultados_df

class DistribucionProbabilidadVector(DistribucionProbabilidad):
    def __init__(self, datos):
        super().__init__(datos)

    def entropia(self):
        # Inicializa un array para guardar la entropía de cada componente
        entropias = np.zeros(3)
        # Calcula la entropía para cada dimensión (columna)
        for i in range(3):  # Asumiendo que tienes 3 dimensiones: X, Y, Z
            p, counts = np.unique(self.datos[:, i], return_counts=True)
            p = counts / len(self.datos)
            entropias[i] = -np.sum(p * np.log(p))
```

```
        return tuple(entropias)

    def calcular_metricas(self):
        # Calcular y redondear cada métrica antes de convertirla en tupla
        media = tuple(np.round(self.media(), 3))
        mediana = tuple(np.round(self.mediana(), 3))
        moda = tuple(np.round(self.moda(), 3))
        media_geometrica = tuple(np.round(self.media_geometrica(), 3))
        rango = tuple(np.round(self.rango(), 3))
        desviacion_estandar = tuple(np.round(self.desviacion_estandar(), 3))
        varianza = tuple(np.round(self.varianza(), 3))
        asimetria = tuple(np.round(self.asimetria(), 3))
        coeficiente_variacion = tuple(np.round(self.coeficiente_variacion(), 3))
        curtosis = tuple(np.round(self.curtosis(), 3))
        entropia = tuple(np.round(self.entropia(), 3))

        resultados = {
            'Media': media,
            'Mediana': mediana,
            'Moda': moda,
            'Media Geométrica': media_geometrica,
            'Rango': rango,
            'Desviación Estándar': desviacion_estandar,
            'Varianza': varianza,
            'Asimetría': asimetria,
            'Coeficiente de Variación': coeficiente_variacion,
            'Curtosis': curtosis,
            'Entropía': entropia
        }

        # Convertir el diccionario a DataFrame de Pandas
        resultados_df = pd.DataFrame([resultados], index=['Valor'])

        # Transponer el DataFrame para hacerlo vertical
        resultados_df = resultados_df.transpose().reset_index()
        resultados_df.columns = ['Métrica', 'Valor']

        return resultados_df
```

[7]:
```
def lyapunov_spectrum_n(datos, window_size):
    """
    Calcula los exponentes de Lyapunov para cualquier serie temporal.

    :param datos: Serie temporal del sistema.
    :param window_size: Tamaño de la ventana para el cálculo del exponente de␣
 ↪Lyapunov.
    :return: Array con los exponentes de Lyapunov.
```

```
    """
    n = len(datos)
    n_windows = n // window_size
    exponentes_lyapunov = []

    # Calcula la derivada numérica usando diferencias finitas
    derivadas = np.diff(datos) / np.diff(np.arange(len(datos)))

    # Añade un valor de derivada al final para mantener el tamaño
    derivadas = np.append(derivadas, derivadas[-1])

    for j in range(n_windows):
        suma_logaritmos_derivadas = 0
        for i in range(window_size):
            derivada = derivadas[j * window_size + i]
            suma_logaritmos_derivadas += np.log(abs(derivada))

        exponente_lyapunov = (1 / window_size) * suma_logaritmos_derivadas
        exponentes_lyapunov.append(exponente_lyapunov)

    return np.array(exponentes_lyapunov)
```

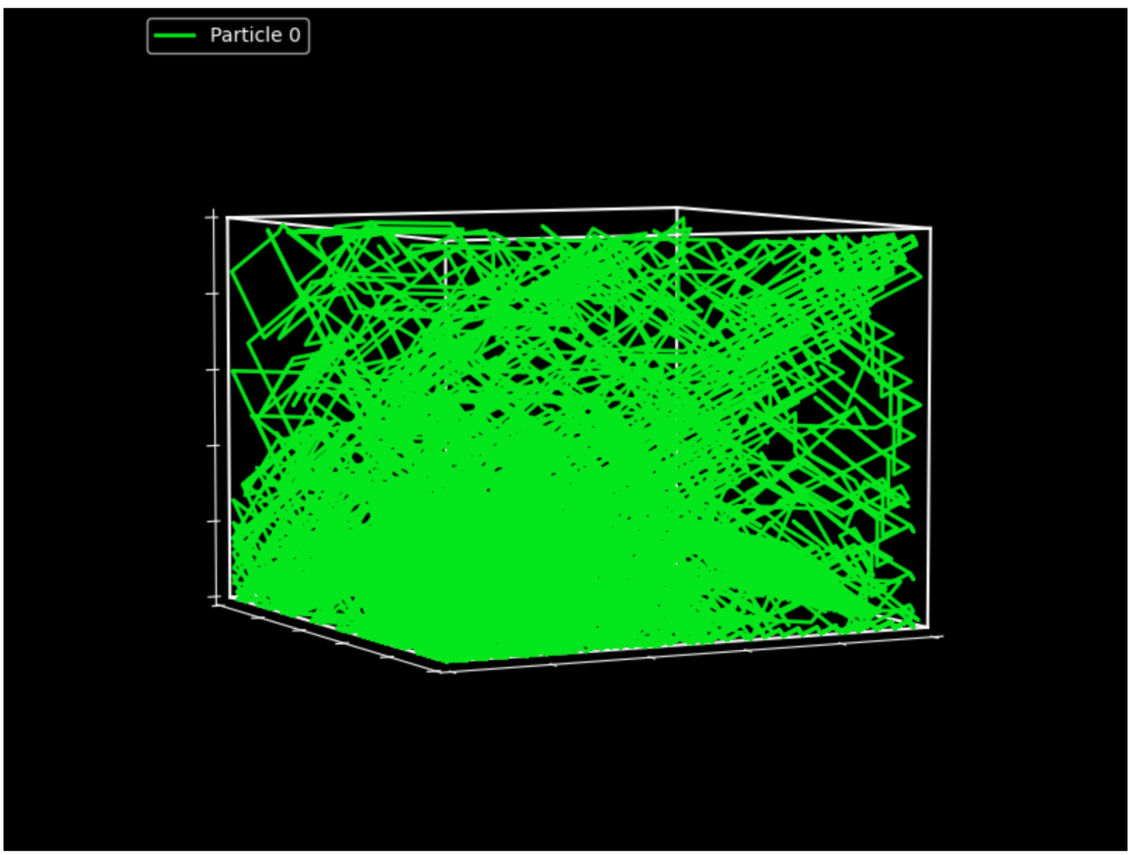
[8]:
```
def lyapunov_spectrum_n(datos, window_size):
    """
    Calcula los exponentes de Lyapunov para cualquier serie temporal.

    :param datos: Serie temporal del sistema.
    :param window_size: Tamaño de la ventana para el cálculo del exponente de␣
 ↪Lyapunov.
    :return: Array con los exponentes de Lyapunov.
    """
    n = len(datos)
    n_windows = n // window_size
    exponentes_lyapunov = []

    # Calcula la derivada numérica usando diferencias finitas
    derivadas = np.diff(datos) / np.diff(np.arange(len(datos)))

    # Añade un valor de derivada al final para mantener el tamaño
    derivadas = np.append(derivadas, derivadas[-1])

    # Evitar valores de cero en las derivadas
    derivadas[derivadas == 0] = np.finfo(float).eps

    for j in range(n_windows):
        suma_logaritmos_derivadas = 0
        for i in range(window_size):
```

```
            derivada = derivadas[j * window_size + i]
            suma_logaritmos_derivadas += np.log(abs(derivada))

        exponente_lyapunov = (1 / window_size) * suma_logaritmos_derivadas
        exponentes_lyapunov.append(exponente_lyapunov)

    return np.array(exponentes_lyapunov)

def plot_lyapunov_spectrum(datos, window_size, folder, metrica):
    """
    Grafica la serie temporal y el espectro de exponentes de Lyapunov.

    :param datos: Serie temporal del sistema.
    :param window_size: Tamaño de la ventana para el cálculo del exponente de␣
 ↪Lyapunov.
    :param folder: Nombre de la carpeta para el título de la gráfica.
    :param metrica: Nombre de la métrica para el título de la gráfica.
    """
    lyapunov_exponents = lyapunov_spectrum_n(datos, window_size)

    fig, ax1 = plt.subplots(figsize=(12, 6))

    color = 'tab:blue'
    ax1.set_xlabel('Tiempo')
    ax1.set_ylabel('Serie Temporal', color=color)
    ax1.scatter(np.arange(len(datos)), datos, color=color, label='Serie␣
 ↪Temporal', marker='o', s=0.1)
    ax1.tick_params(axis='y', labelcolor=color)

    ax2 = ax1.twinx()
    color = 'tab:red'
    ax2.set_ylabel('Exponente de Lyapunov', color=color)
    ax2.plot(np.arange(len(lyapunov_exponents)) * window_size + window_size //␣
 ↪2, lyapunov_exponents, color=color, label='Exponente de Lyapunov')
    ax2.tick_params(axis='y', labelcolor=color)

    fig.suptitle(f'Serie Temporal y Espectro de Exponentes de Lyapunov de␣
 ↪{folder} - {metrica}')
    fig.tight_layout()
    plt.show()

    return lyapunov_exponents
```

[9]:
```
def dimension_kaplan_yorke(exponentes_lyapunov):
    """
    Calcula la dimensión de Kaplan-Yorke dada una lista de exponentes de␣
 ↪Lyapunov.
```

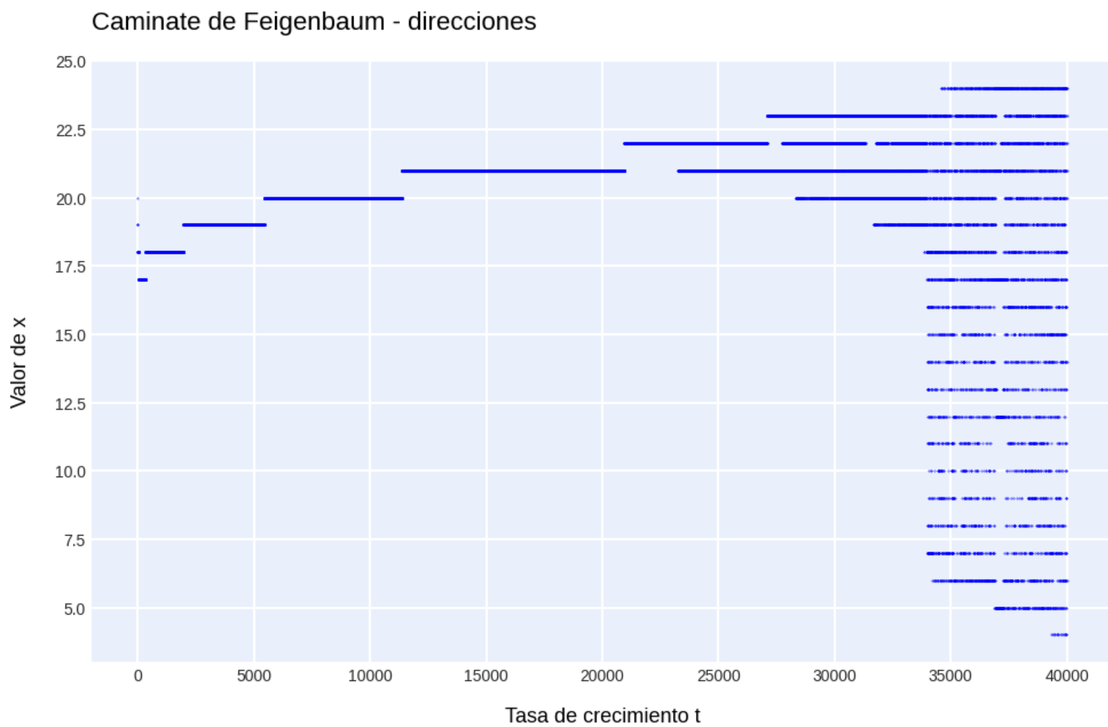
```
    :param exponentes_lyapunov: Lista de exponentes de Lyapunov.
    :return: Dimensión de Kaplan-Yorke.
    """
    exponentes_lyapunov = sorted(exponentes_lyapunov, reverse=True)

    if len(exponentes_lyapunov) < 3:
        raise ValueError("Se requieren al menos 3 exponentes de Lyapunov para␣
↪calcular la dimensión de Kaplan-Yorke con esta fórmula.")

    suma = sum(exponentes_lyapunov[:2])

    if exponentes_lyapunov[2] == 0:
        return float('inf')  # O maneja el caso de manera adecuada
    else:
        dk_y = 2 + (suma / abs(exponentes_lyapunov[2]))

    return dk_y
```

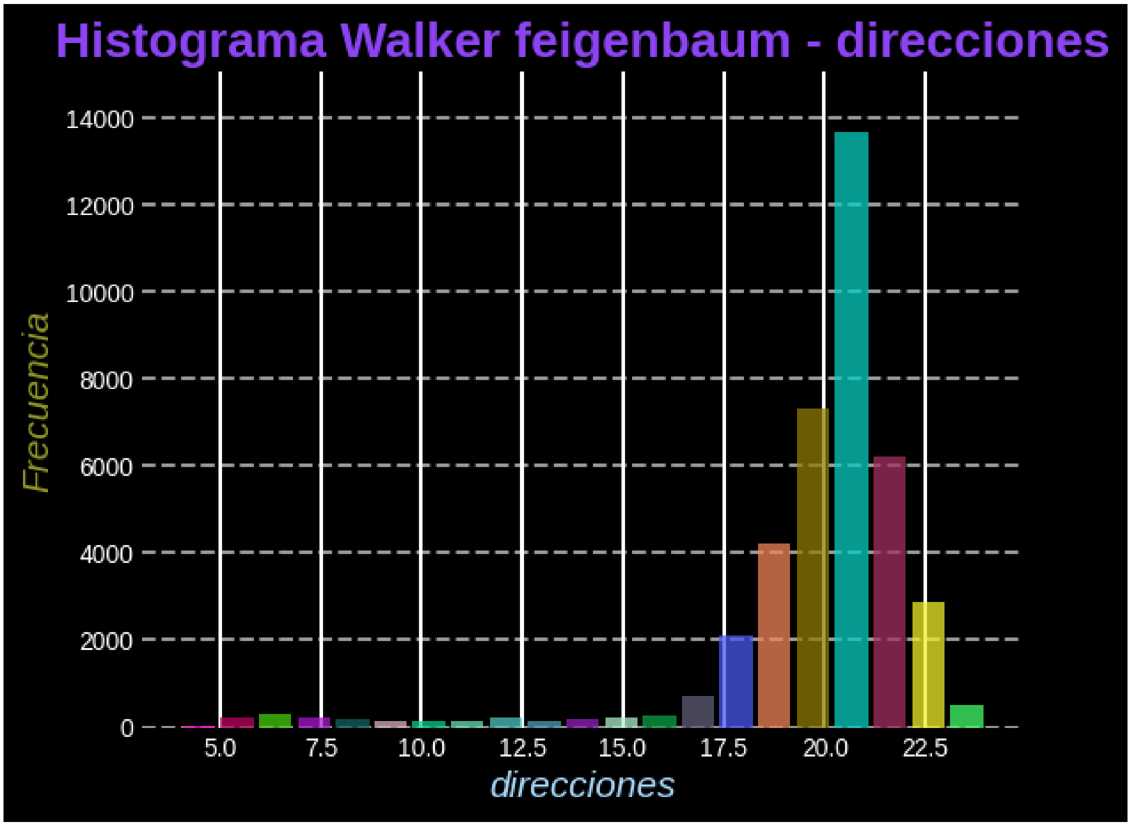
1.3 Caminante de Feigenbaum



1.3.1 Dirección

[10]:
```
modelo = "feigenbaum"
metrica = "direcciones"
folder = "Feigenbaum"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {convertir_camelCase(modelo)} -␣
↪{metrica}", metrica, "Frecuencia", criterio="secreto")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```





Interpretación de las Métricas Media (20.068): La media es el valor promedio de las direcciones seleccionadas por el caminante. En este caso, la media de 20.068 indica que, en promedio, las direcciones elegidas están alrededor del índice 20 en el rango de posibles direcciones.

Mediana (21.0): La mediana es el valor que divide el conjunto de datos en dos partes iguales. En este caso, la mediana de 21.0 sugiere que la mitad de las direcciones seleccionadas son menores o iguales a 21 y la otra mitad son mayores o iguales a 21.

Moda (21, 13678): La moda es el valor que aparece con más frecuencia en el conjunto de datos. Aquí, la dirección 21 es la más frecuente, seleccionada 13,678 veces. Esto indica una preferencia significativa del caminante por moverse en la dirección 21.

Desviación Estándar (2.965): La desviación estándar mide la dispersión de los valores alrededor de la media. Una desviación estándar de 2.965 sugiere que las direcciones seleccionadas por el caminante tienden a variar en aproximadamente ± 3 direcciones alrededor de la media (20.068). Esto indica una cierta concentración alrededor de la media, pero con variabilidad.

Varianza (8.788): La varianza es el cuadrado de la desviación estándar y proporciona una medida de la dispersión de los datos. Una varianza de 8.788 confirma la observación de la desviación estándar, indicando una variabilidad moderada en las direcciones seleccionadas.

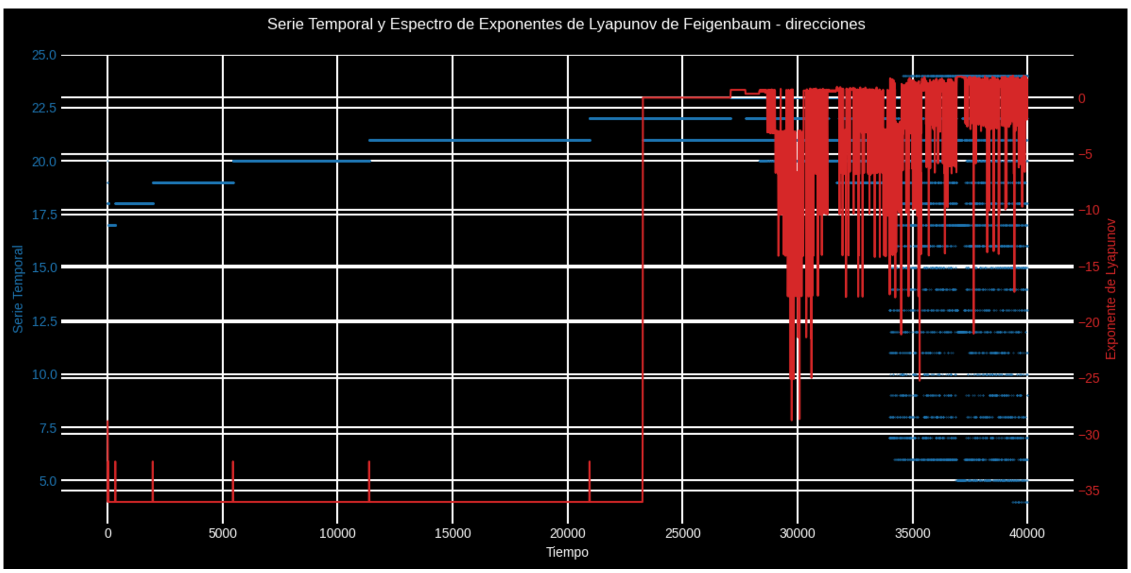
Asimetría (-2.969): La asimetría mide la simetría de la distribución de los datos. Un valor de asimetría de -2.969 indica que la distribución está sesgada hacia la izquierda. Esto significa que hay una cola más larga en el lado izquierdo de la distribución, sugiriendo que hay algunas direcciones de menor índice que se seleccionan con menor frecuencia, pero de manera más dispersa.

Curtosis (10.539): La curtosis mide la "puntiagudez" de la distribución de los datos. Un valor de curtosis de 10.539 indica que la distribución es muy leptocúrtica, lo que significa que tiene una alta concentración de valores en torno a la media y colas más largas. Esto sugiere que, además de la moda, hay una concentración significativa de valores alrededor de la media y la mediana, con menos valores extremos.

Entropía (1.996): La entropía mide la incertidumbre o aleatoriedad en los datos. Una entropía de 1.996 indica un nivel moderado de incertidumbre en la selección de direcciones, lo que sugiere que aunque hay una preferencia por ciertas direcciones (como la dirección 21), hay también variabilidad en la selección de otras direcciones.

Exponentes de Lyapounov y Kaplan York

[11]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
    ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```




```
La dimension de Kaplan York es: 4.048835893117703
```

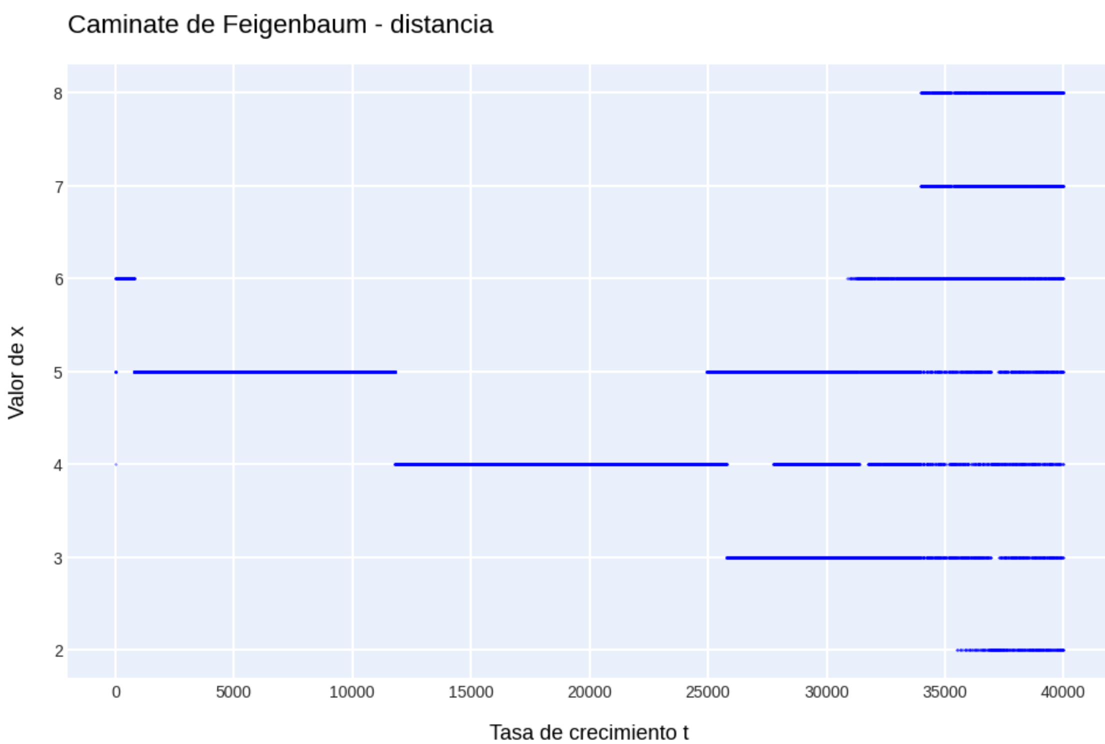
La gráfica muestra la evolución temporal de las direcciones seleccionadas por el caminante aleatorio, superpuesta con el espectro de exponentes de Lyapunov.

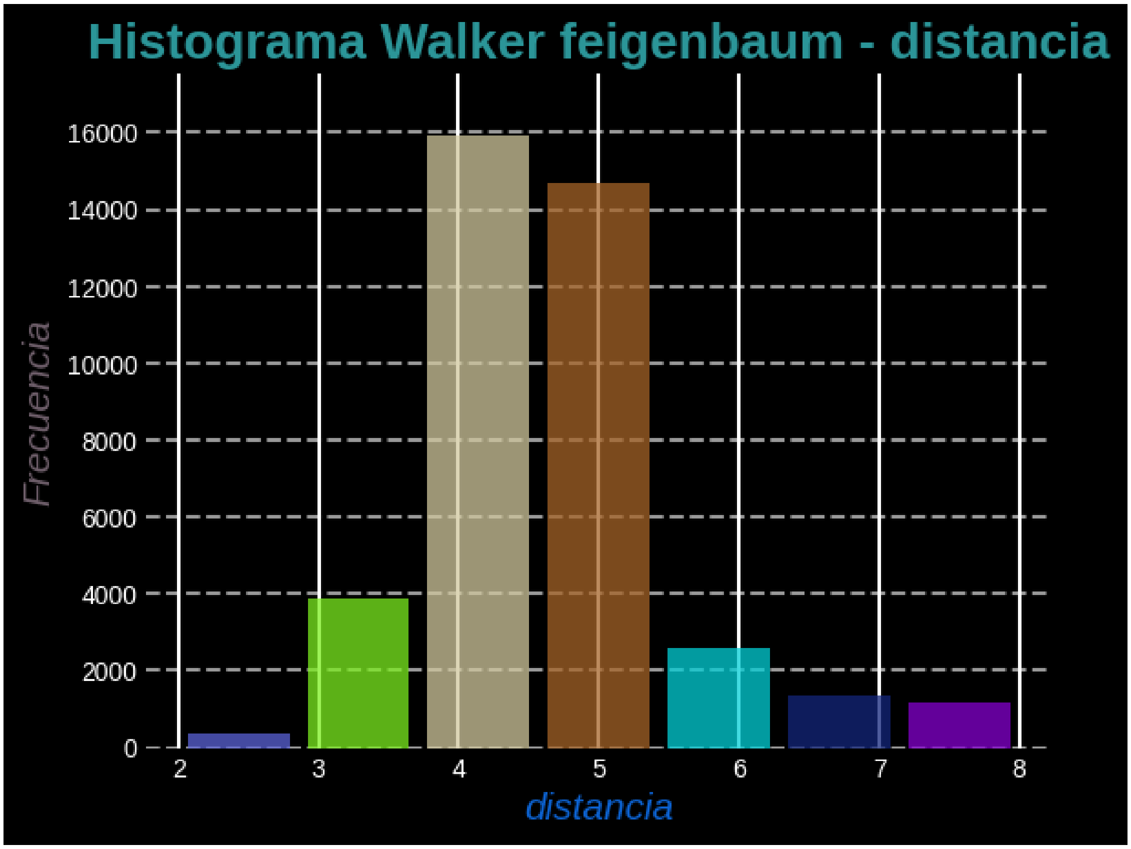
- **Serie Temporal**: Se observa una tendencia de las direcciones alrededor de valores específicos (mayores a 15), con un comportamiento más errático hacia el final de la serie.
- **Exponentes de Lyapunov**: Los exponentes de Lyapunov negativos indican estabilidad local y sensibilidad a las condiciones iniciales. La variabilidad en estos valores a lo largo del tiempo refleja la naturaleza caótica del sistema.

- **Dimensión de Kaplan Yorke**: 4.048835893117703. Este valor sugiere una complejidad en la dinámica del sistema, con una alta dimensión fractal que es típica en sistemas caóticos.

1.3.2 Distancia

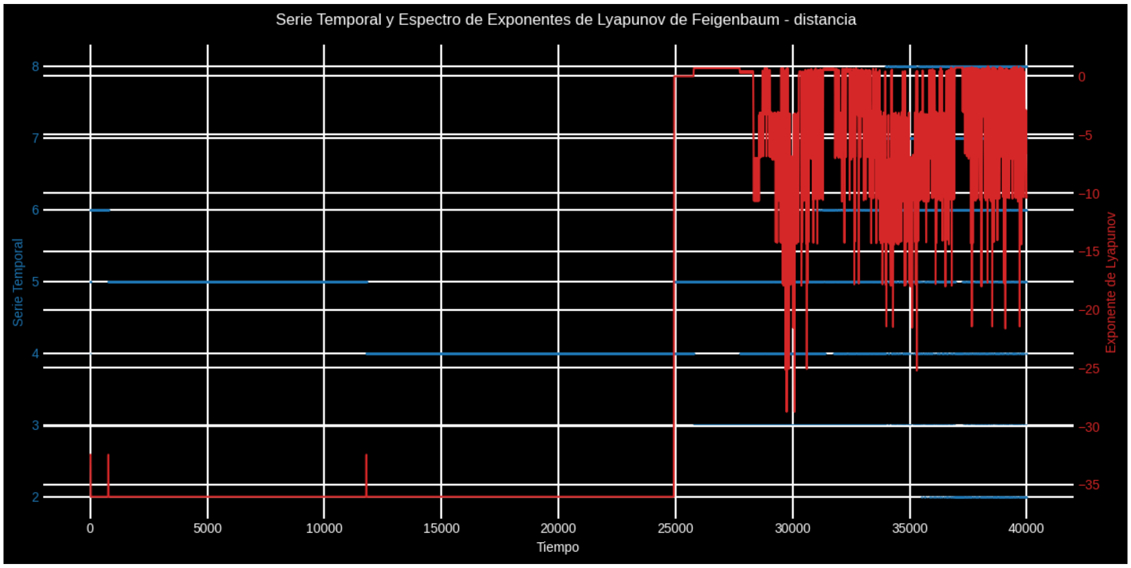
[12]:
```
metrica = "distancia"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
  ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {convertir_camelCase(modelo)} -␣
  ↪{metrica}", metrica, "Frecuencia", criterio="secreto")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```





Exponentes de Lyapounov Y Kaplan Yorke

[13]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
  ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```



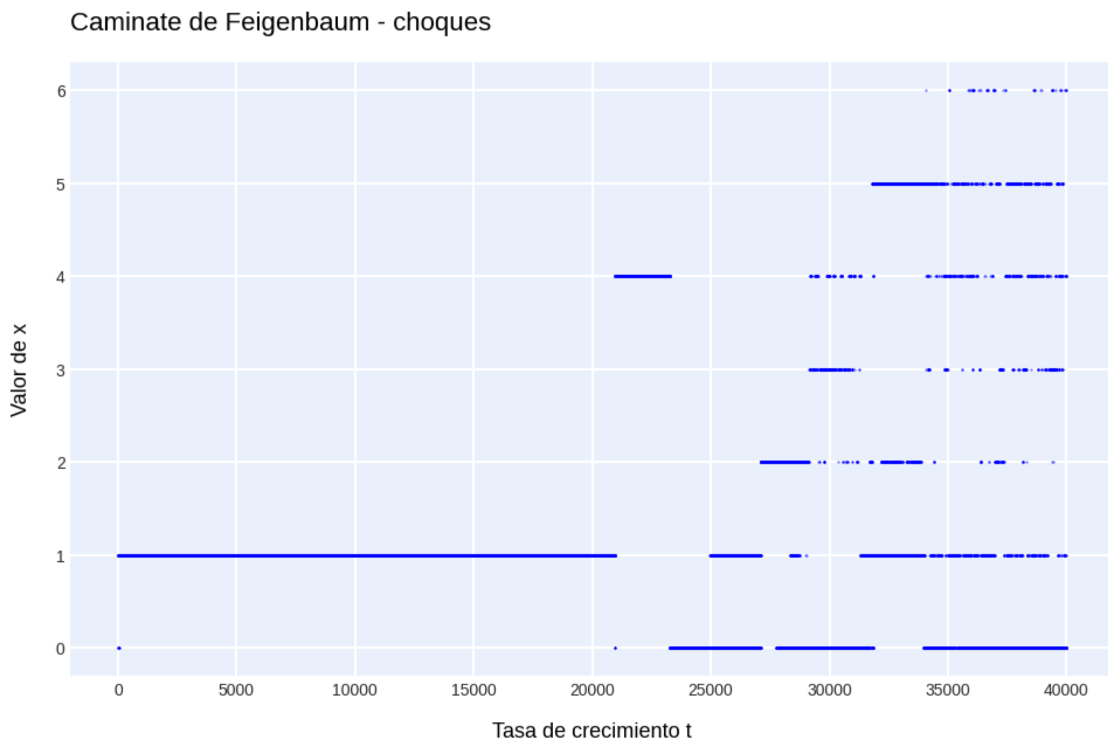

```
La dimension de Kaplan York es: 4.155383905224292
```

Esta gráfica ilustra la evolución de las distancias recorridas en cada paso por el caminante aleatorio, junto con el espectro de exponentes de Lyapunov.

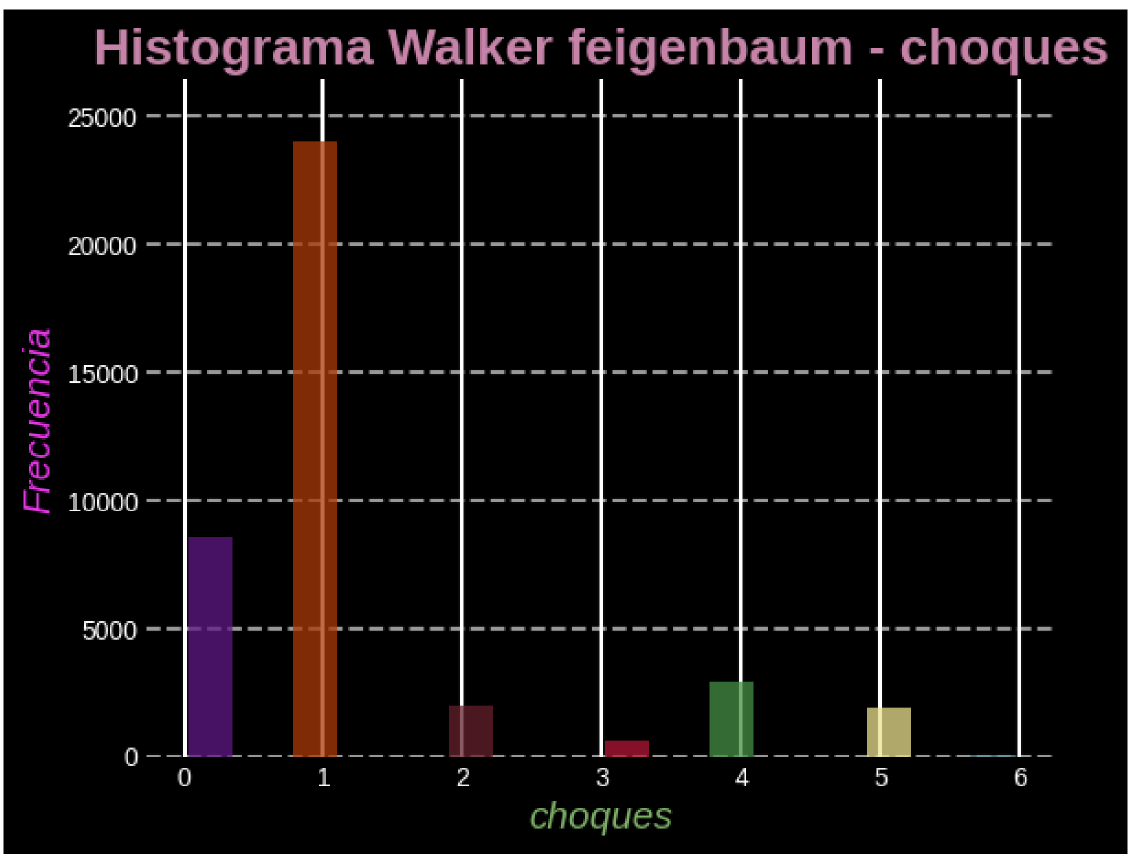
- **Serie Temporal**: Las distancias se concentran en valores bajos (1 a 4), lo que sugiere que el caminante realiza pequeños desplazamientos con más frecuencia.
- **Exponentes de Lyapunov**: Al igual que en la gráfica de direcciones, los exponentes negativos predominan, indicando regiones de estabilidad y comportamiento caótico.
- **Dimensión de Kaplan Yorke**: 4.15538905224292. Este valor refuerza la idea de una dinámica compleja y caótica en el movimiento del caminante.

1.3.3 Choques

[14]:
```
metrica = "choques"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
 ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {convertir_camelCase(modelo)} -␣
 ↪{metrica}", metrica, "Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```



Caminate de Feigenbaum - choques
Valor de x
Tasa de crecimiento t
0
1
2
3
4
5
6
0
5000
10000
15000
20000
25000
30000
35000
40000



Histograma Walker feigenbaum - choques
Frecuencia
choques
0
5000
10000
15000
20000
25000
0
1
2
3
4
5
6

Interpretación de las Métricas Media (1.282): La media es el valor promedio de los choques que experimenta el caminante. En este caso, la media de 1.282 indica que, en promedio, el caminante choca con las paredes del cubo aproximadamente 1.28 veces por movimiento.

Mediana (1.0): La mediana es el valor que divide el conjunto de datos en dos partes iguales. En este caso, la mediana de 1.0 sugiere que la mitad de los choques son menores o iguales a 1 y la otra mitad son mayores o iguales a 1.

Moda (1, 24049): La moda es el valor que aparece con más frecuencia en el conjunto de datos. Aquí, el número de choques de 1 es el más frecuente, ocurriendo 24,049 veces. Esto indica una fuerte tendencia del caminante a chocar con las paredes del cubo una vez por movimiento.

Desviación Estándar (1.304): La desviación estándar mide la dispersión de los valores alrededor de la media. Una desviación estándar de 1.304 sugiere que el número de choques tiende a variar en aproximadamente ± 1.3 alrededor de la media (1.282). Esto indica una cierta variabilidad en el número de choques, aunque la mayoría están cerca de la media.

Varianza (1.701): La varianza es el cuadrado de la desviación estándar y proporciona una medida de la dispersión de los datos. Una varianza de 1.701 confirma la observación de la desviación estándar, indicando una variabilidad moderada en el número de choques.

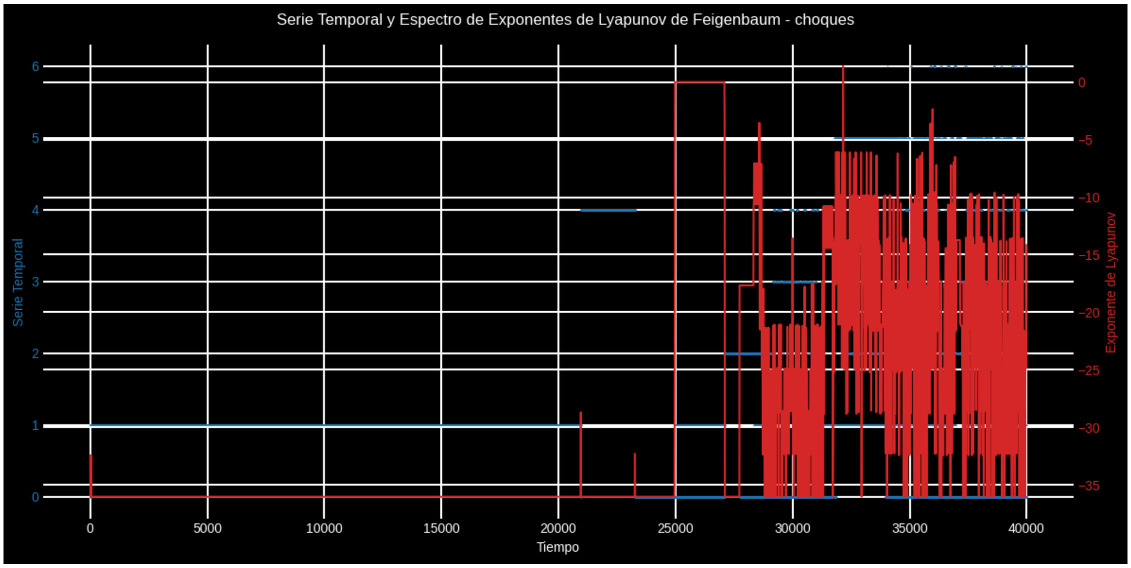
Asimetría (1.678): La asimetría mide la simetría de la distribución de los datos. Un valor de asimetría de 1.678 indica que la distribución está sesgada hacia la derecha. Esto significa que hay una cola más larga en el lado derecho de la distribución, sugiriendo que hay algunos movimientos con un mayor número de choques que ocurren con menor frecuencia.

Curtosis (2.073): La curtosis mide la "puntiagudez" de la distribución de los datos. Un valor de curtosis de 2.073 indica que la distribución es leptocúrtica, lo que significa que tiene una concentración de valores en torno a la media y colas más largas. Esto sugiere que, además de la moda, hay una concentración significativa de valores alrededor de la media y la mediana, con menos valores extremos.

Entropía (1.192): La entropía mide la incertidumbre o aleatoriedad en los datos. Una entropía de 1.192 indica un nivel moderado de incertidumbre en la ocurrencia de choques, lo que sugiere que aunque hay una preferencia por ciertos números de choques (como 1 choque), hay también variabilidad en la ocurrencia de otros números de choques.

Exponentes de Lyapounov y Kaplan Yorke

[15]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,
    metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```



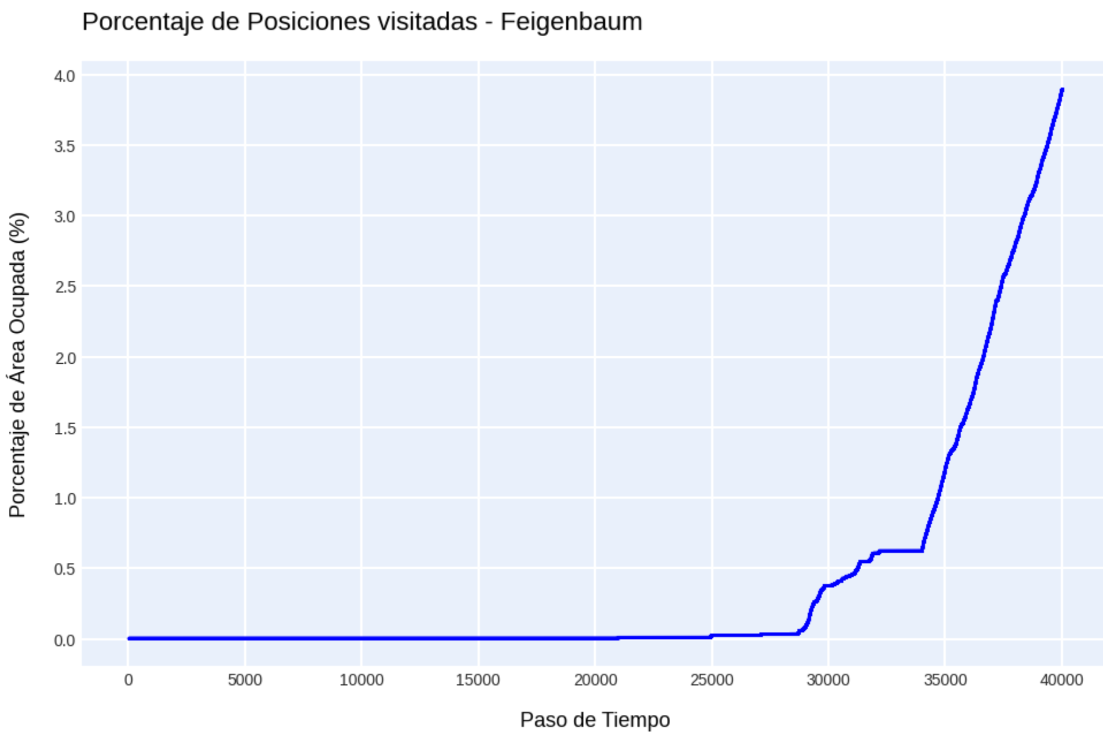

```
La dimension de Kaplan York es: inf
```

La gráfica muestra el comportamiento de los choques del caminante con las paredes del cubo, en combinación con los exponentes de Lyapunov.

- **Serie Temporal**: La mayoría de los datos indican choques con la pared etiquetada como '0' (sin choque), con algunos episodios de choques con otras paredes, especialmente hacia el final del tiempo.
- **Exponentes de Lyapunov**: Los exponentes negativos y su variabilidad reflejan el carácter caótico del sistema y su sensibilidad a las condiciones iniciales.
- **Dimensión de Kaplan Yorke**: Inf. Este resultado indica un nivel muy alto de complejidad y caos en el sistema, lo que podría estar relacionado con la alta frecuencia y variabilidad de los choques.

1.3.4 Posiciones

[16]:
```
metrica = "posiciones"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ";")
graficar_posicionesOcupadas(49, walker_feig, title=f"Porcentaje de Posiciones␣
 ↪visitadas - {folder}")
ex = DistribucionProbabilidadVector(np.array(walker_feig))
res = ex.calcular_metricas()
```



Interpretación de las Métricas Media (19.862, 7.186, 36.397) : La media sugiere que, en promedio, las posiciones visitadas por el caminante son bastante variadas en el espacio tridimensional, con un sesgo hacia posiciones específicas dentro del cubo.

Mediana (20.0, 0.0, 49.0): La mediana indica que la mitad de las posiciones visitadas se encuentran por debajo de estos valores, sugiriendo una concentración de posiciones visitadas en ciertas áreas del cubo.

Moda ([20, 0, 49], [22135, 24255, 22538]): La moda muestra que hay posiciones específicas (como [20, 0, 49]) que son visitadas con mucha más frecuencia que otras, lo que indica una preferencia o un patrón repetitivo en el movimiento del caminante.

Rango (49, 49, 49): El rango de 49 en cada dimensión indica que el caminante cubre prácticamente todo el cubo, sugiriendo que no hay restricciones significativas en su movimiento y que eventualmente explora la mayoría del espacio disponible.

Desviación Estándar (12.46, 13.25, 18.253): La desviación estándar muestra una variabilidad considerable en las posiciones visitadas, con la mayor variabilidad en la tercera dimensión, lo que sugiere una exploración más dispersa en esa dirección.

Varianza (155.256, 175.568, 333.158): La varianza refuerza la observación de la desviación estándar, mostrando una dispersión significativa en las posiciones visitadas, especialmente en la tercera dimensión.

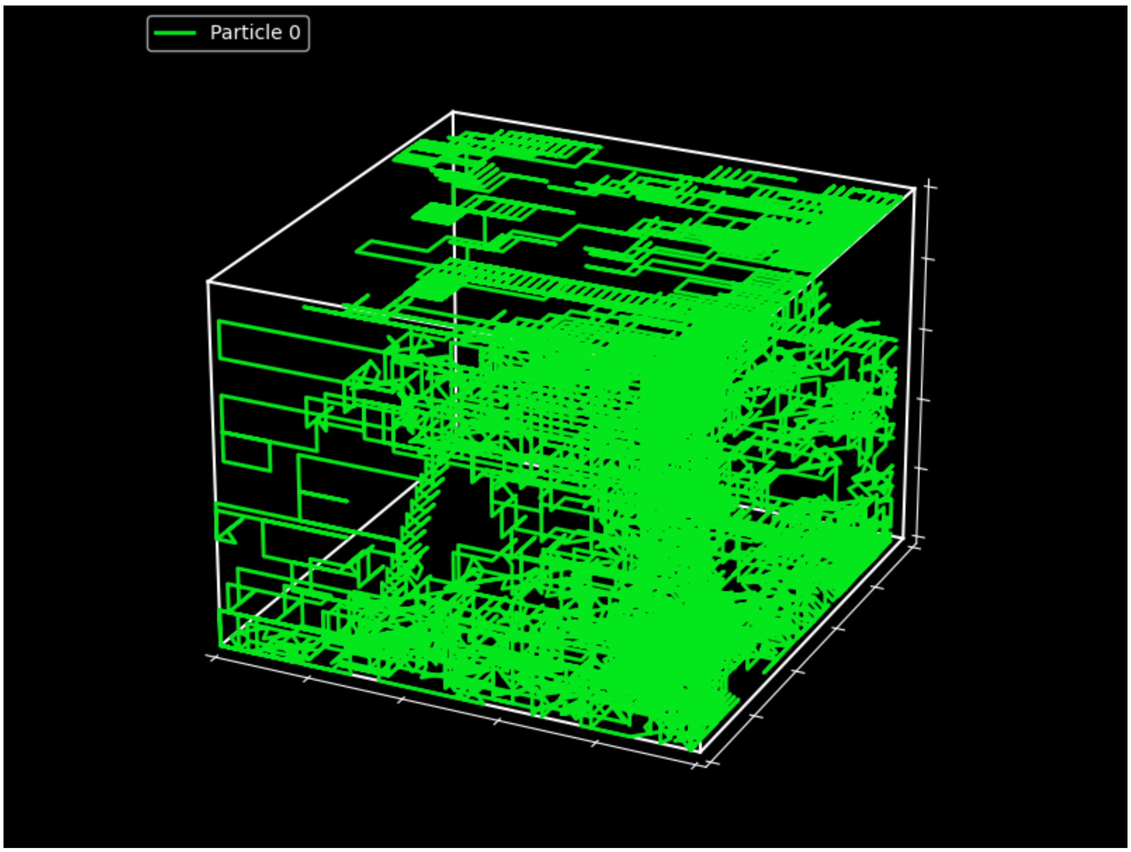
Asimetría (0.662, 1.898, -1.072) La asimetría sugiere que la distribución de posiciones visitadas tiene diferentes sesgos en cada dimensión, con una distribución más equilibrada en la primera y segunda dimensiones, y una distribución sesgada hacia valores menores en la tercera dimensión.

Coeficiente de Variación (0.627, 1.844, 0.501): El coeficiente de variación indica que hay una relativa consistencia en la exploración a lo largo de las dimensiones, con la segunda dimensión mostrando más variabilidad relativa.

Curtosis (0.63, 2.51, -0.506): La curtosis revela que las posiciones visitadas tienen distribuciones variadas, con una distribución más plana en la primera y tercera dimensiones, y una distribución más puntiaguda en la segunda dimensión.

Entropía (2.116, 1.876, 2.014): La entropía indica un nivel moderado de incertidumbre en la exploración de posiciones, con cierta aleatoriedad en el movimiento del caminante dentro del cubo.

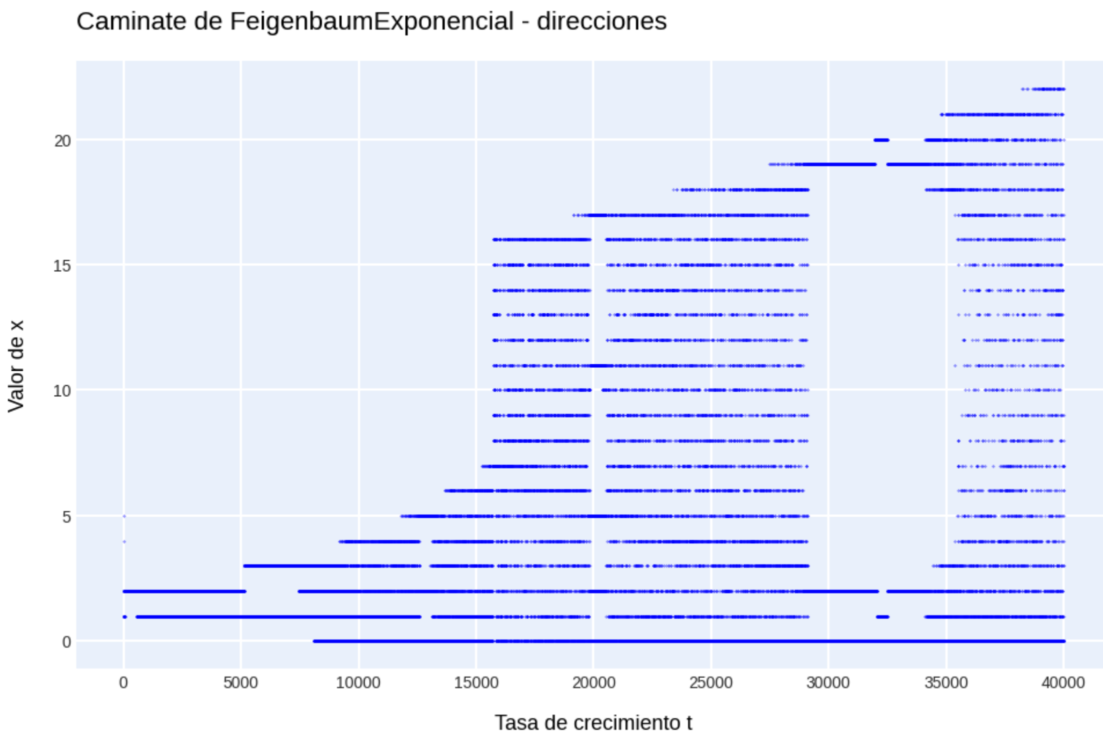
1.4 Caminante de Feigenbaum Exponencial

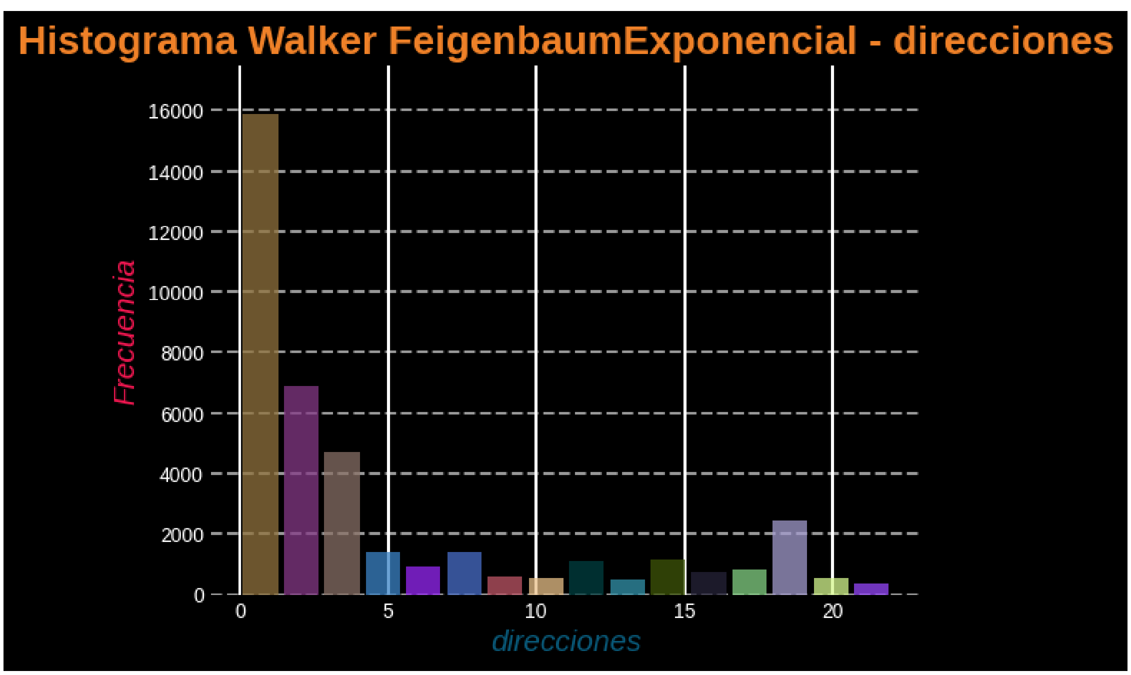


1.4.1 Direcciones

[17]:
```
modelo = "feigenbaumexponencial"
metrica = "direcciones"
folder = "FeigenbaumExponencial"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
```

```
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
 ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```





Interpretación de las Métricas Media (4.904): La media indica que, en promedio, las direcciones seleccionadas por el caminante están alrededor del índice 4.904.

Mediana (2.0): La mediana sugiere que la mitad de las direcciones seleccionadas son menores o iguales a 2 y la otra mitad son mayores o iguales a 2.

Moda (0, 8224): La moda muestra que la dirección 0 es la más frecuente, seleccionada 8,224 veces, lo que indica una preferencia significativa del caminante por moverse en esta dirección.

Desviación Estándar (6.123: La desviación estándar de 6.123 indica que las direcciones seleccionadas varían considerablemente alrededor de la media.

Varianza (37.497): La varianza de 37.497 confirma una alta dispersión en las direcciones seleccionadas.

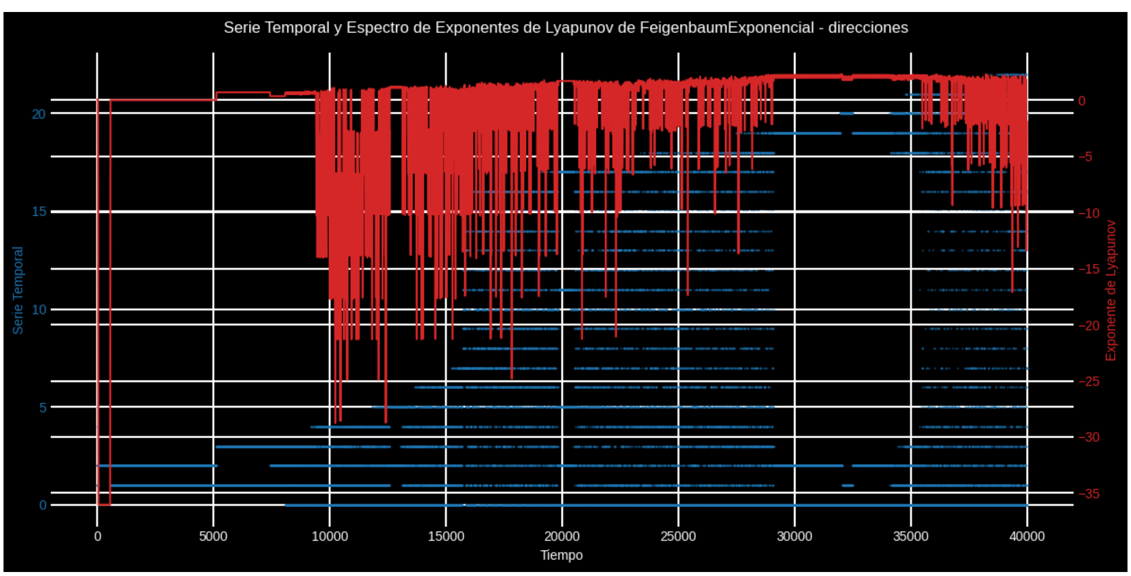
Asimetría (1.354): La asimetría de 1.354 sugiere que la distribución está sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (0.436): La curtosis de 0.436 indica que la distribución es más plana en comparación con una distribución normal, sugiriendo menos valores extremos.

Entropía (2.492): La entropía de 2.492 indica un alto nivel de incertidumbre o aleatoriedad en las direcciones seleccionadas por el caminante.

EXponentes de Lyapounov y Kaplan Yorke

[18]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
    ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```



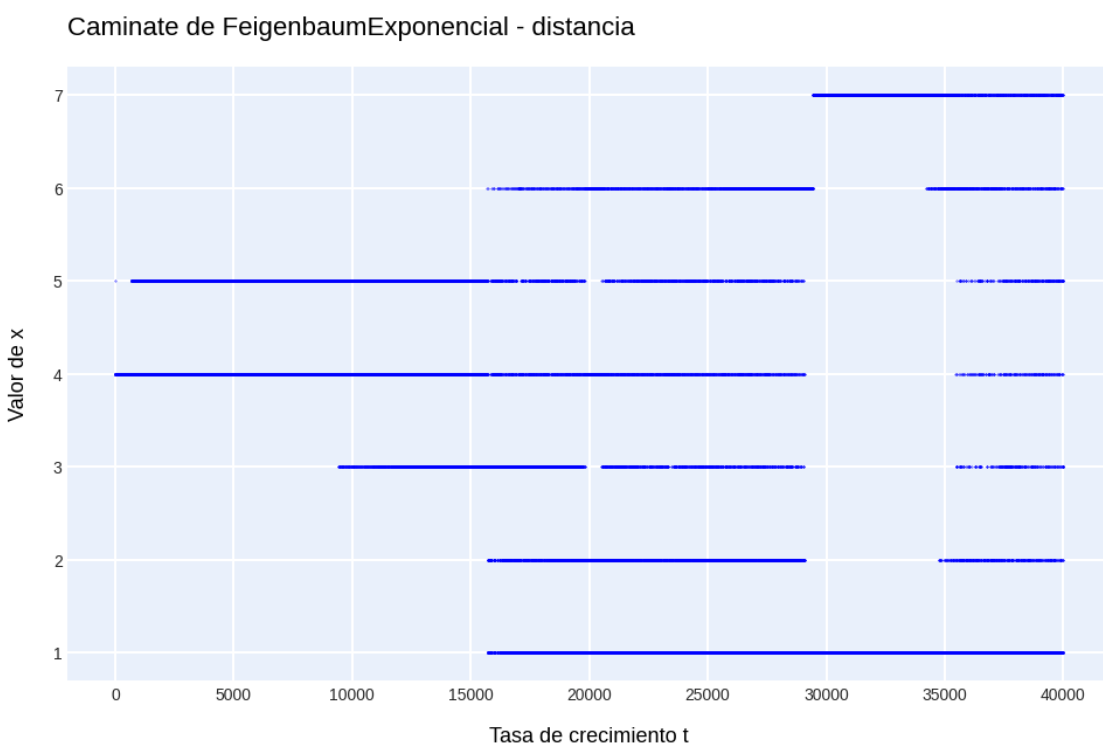
```
La dimension de Kaplan York es: 4.0332396702669016
```

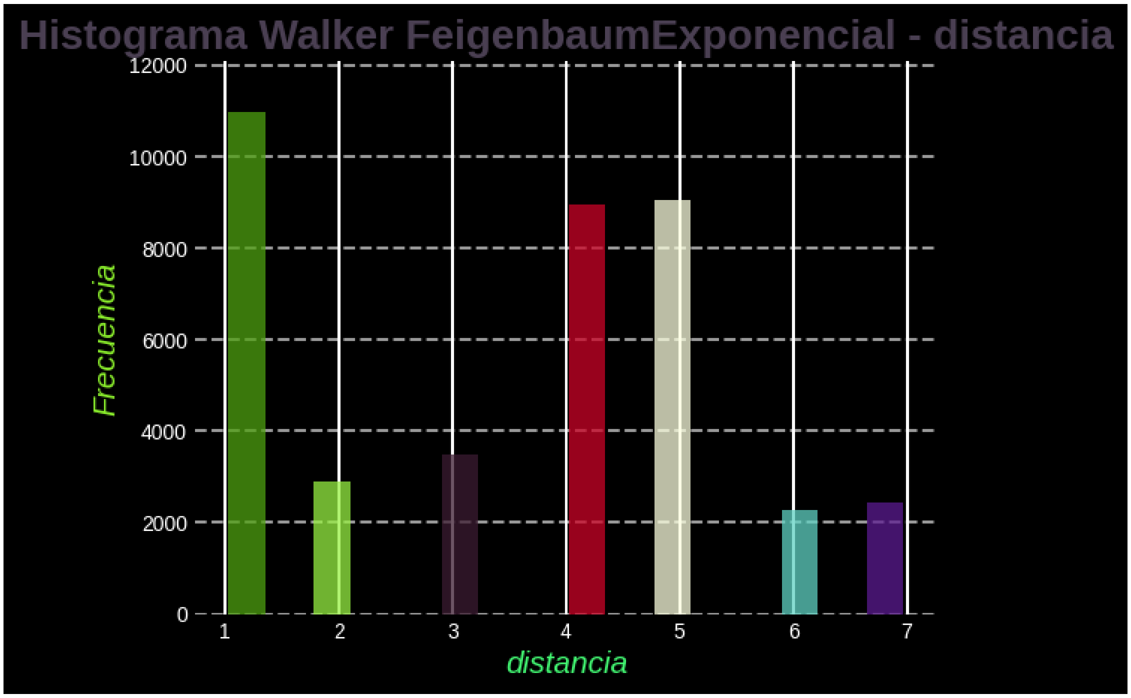
La gráfica muestra la evolución temporal de las direcciones seleccionadas por el caminante aleatorio, superpuesta con el espectro de exponentes de Lyapunov.

- **Serie Temporal**: Se observa una tendencia de las direcciones alrededor de valores específicos (mayores a 15), con un comportamiento más errático hacia el final de la serie.
- **Exponentes de Lyapunov**: Los exponentes de Lyapunov negativos indican estabilidad local y sensibilidad a las condiciones iniciales. La variabilidad en estos valores a lo largo del tiempo refleja la naturaleza caótica del sistema.
- **Dimensión de Kaplan Yorke**: 4.048835893117703. Este valor sugiere una complejidad en la dinámica del sistema, con una alta dimensión fractal que es típica en sistemas caóticos.

1.4.2 Distancia

[19]:
```
metrica = "distancia"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
 ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
 ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```





Interpretación de las Métricas Media (3.467): La media indica que, en promedio, el caminante avanza aproximadamente 3.467 unidades por movimiento.

Mediana (4.0): La mediana sugiere que la mitad de las distancias recorridas son menores o iguales a 4 y la otra mitad son mayores o iguales a 4.

Moda (1, 10985): La moda muestra que la distancia de 1 unidad es la más frecuente, seleccionada 10,985 veces, lo que indica una fuerte preferencia por esta distancia.

Desviación Estándar (1.887): La desviación estándar de 1.887 sugiere una variabilidad moderada en las distancias recorridas.

Varianza (3.563): La varianza de 3.563 confirma la observación de la desviación estándar, indicando una dispersión moderada en las distancias recorridas.

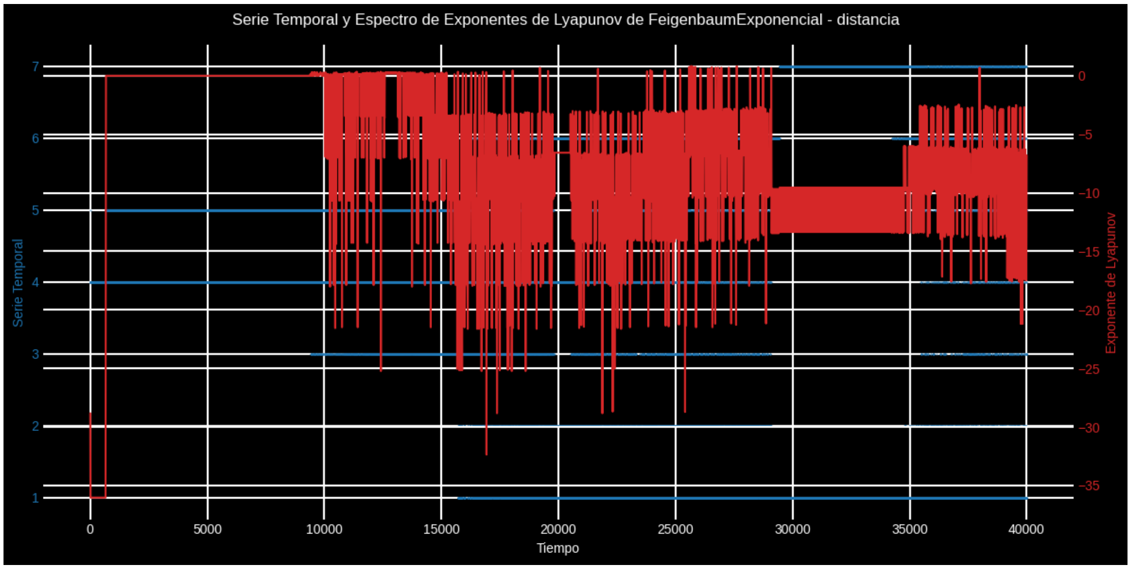
Asimetría (0.015): La asimetría cercana a 0 indica que la distribución de las distancias es bastante simétrica.

Curtosis (1.141): La curtosis de 1.141 sugiere una distribución ligeramente más plana en comparación con una distribución normal, indicando menos valores extremos.

Entropía (1.76): La entropía de 1.76 indica un nivel moderado de incertidumbre en la selección de distancias.

1.4.3 Exponentes de Lyapounov y Kaplan Yorke

[20]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```




```
La dimension de Kaplan York es: 4.0
```

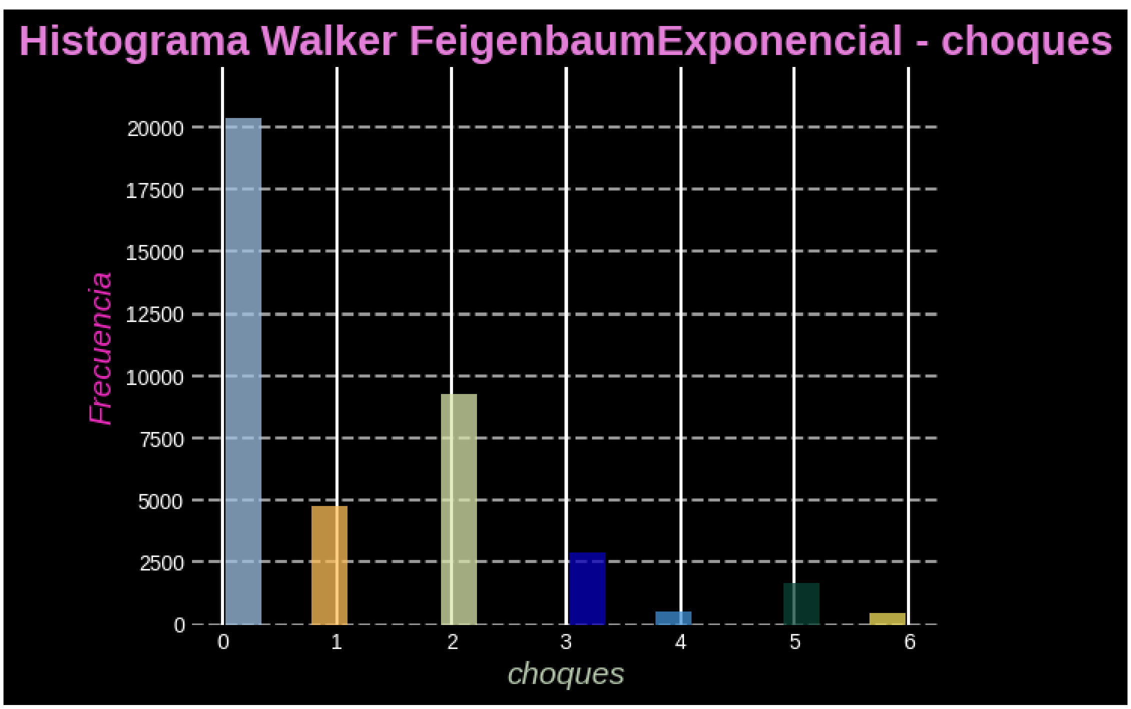
Esta gráfica ilustra la evolución de las distancias recorridas en cada paso por el caminante aleatorio, junto con el espectro de exponentes de Lyapunov.

- **Serie Temporal**: Las distancias se concentran en valores bajos (1 a 4), lo que sugiere que el caminante realiza pequeños desplazamientos con más frecuencia.
- **Exponentes de Lyapunov**: Al igual que en la gráfica de direcciones, los exponentes negativos predominan, indicando regiones de estabilidad y comportamiento caótico.
- **Dimensión de Kaplan Yorke**: 4.15538905224292. Este valor refuerza la idea de una dinámica compleja y caótica en el movimiento del caminante.

1.4.4 Choques

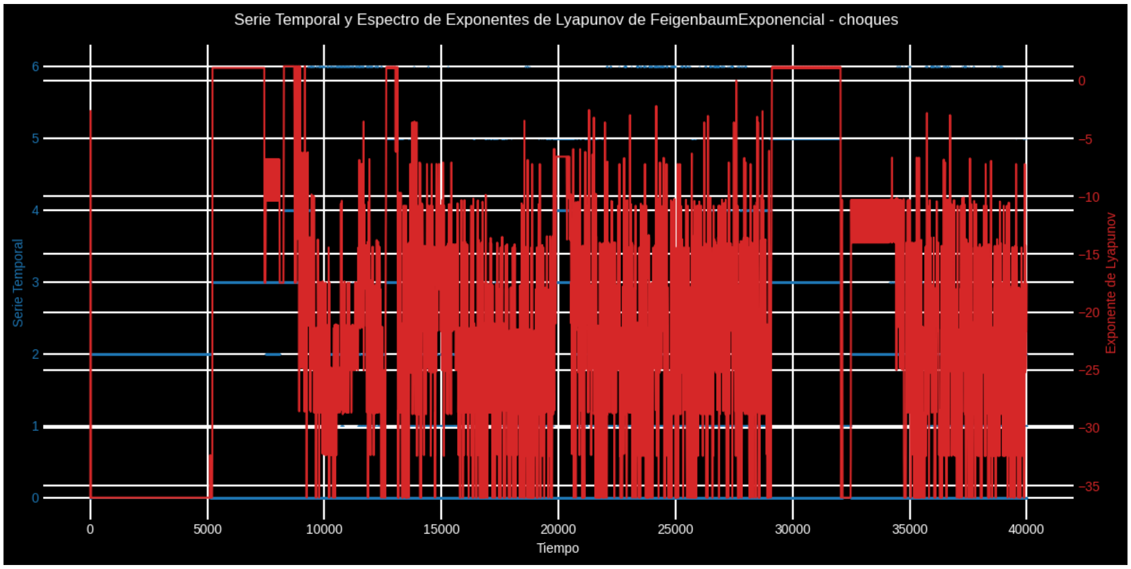
[21]:
```
metrica = "choques"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
```

```
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```



1.4.5 Exponentes de lyapounov y Kaplan Yorke

[22]:
```python
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```




```
La dimension de Kaplan York es: 4.0
```

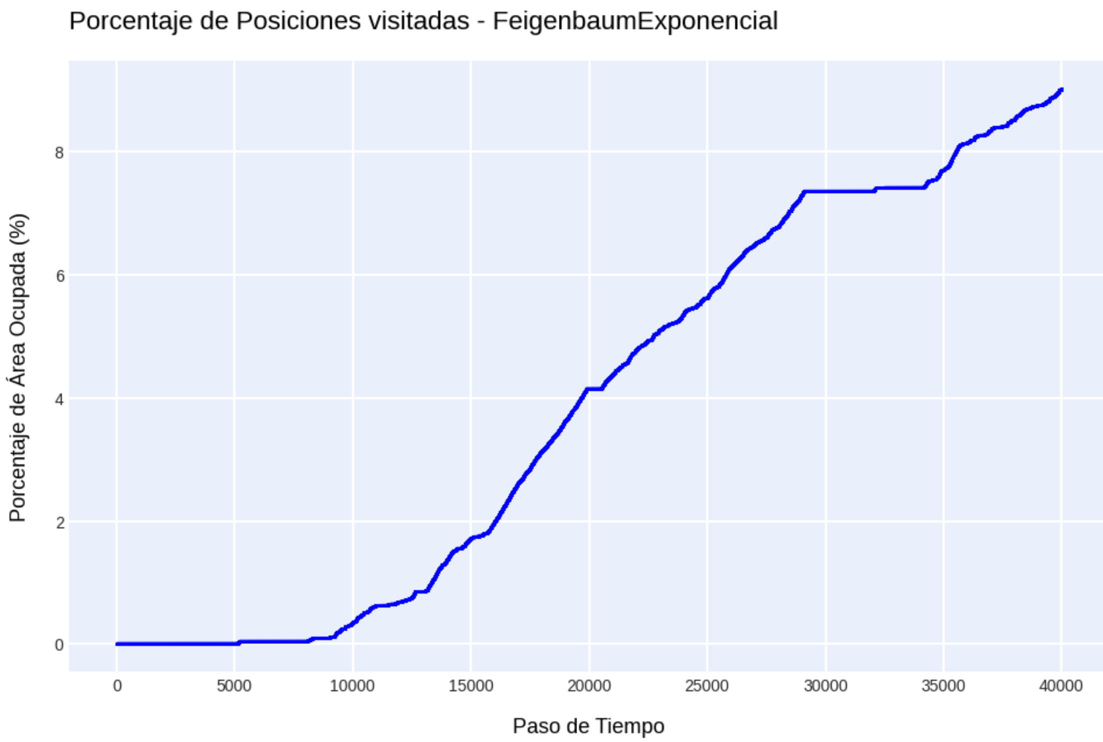
La gráfica muestra el comportamiento de los choques del caminante con las paredes del cubo, en combinación con los exponentes de Lyapunov.

- **Serie Temporal**: La mayoría de los datos indican choques con la pared etiquetada como '0' (sin choque), con algunos episodios de choques con otras paredes, especialmente hacia el final del tiempo.
- **Exponentes de Lyapunov**: Los exponentes negativos y su variabilidad reflejan el carácter caótico del sistema y su sensibilidad a las condiciones iniciales.
- **Dimensión de Kaplan Yorke**: Inf. Este resultado indica un nivel muy alto de complejidad y caos en el sistema, lo que podría estar relacionado con la alta frecuencia y variabilidad de los choques.

1.4.6 Posiciones

[23]:
```python
metrica = "posiciones"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ";")

graficar_posicionesOcupadas(49, walker_feig, title=f"Porcentaje de Posiciones␣
↪visitadas - {folder}")
ex = DistribucionProbabilidadVector(np.array(walker_feig))
```

```
res = ex.calcular_metricas()
```



Interpretación de las Métricas Media (33.164, 25.167, 27.957): La media indica que, en promedio, las posiciones visitadas por el caminante están alrededor de estos valores en el espacio tridimensional, con un sesgo hacia ciertas áreas del cubo.

Mediana (44.0, 24.0, 29.0): La mediana sugiere que la mitad de las posiciones visitadas se encuentran por debajo de estos valores, indicando una concentración en ciertas áreas del cubo.

Moda ([49, 49, 29], [14608, 10551, 9427]): La moda muestra que ciertas posiciones específicas son visitadas con mucha más frecuencia que otras, lo que indica una fuerte preferencia o patrón repetitivo en el movimiento del caminante.

Desviación Estándar (19.52, 21.569, 16.843): La desviación estándar indica una considerable variabilidad en las posiciones visitadas, especialmente en la segunda dimensión.

Varianza (381.046, 465.225, 283.703): La varianza confirma la alta dispersión en las posiciones visitadas, con una mayor dispersión en la segunda dimensión.

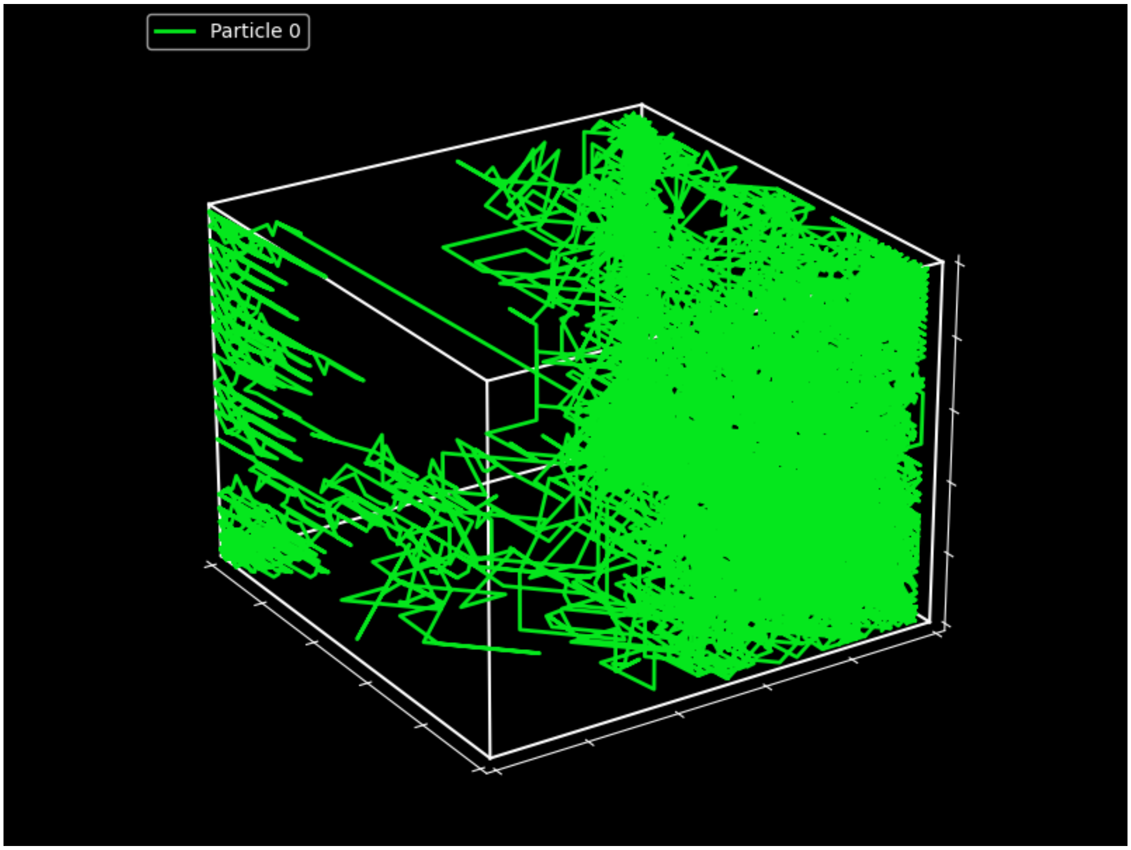
Asimetría (-0.808, 0.0, -0.4): La asimetría sugiere que la distribución de las posiciones visitadas tiene diferentes sesgos en cada dimensión, con una distribución más equilibrada en la segunda dimensión y distribuciones sesgadas hacia valores menores en las otras dos dimensiones.

Coeficiente de Variación (0.589, 0.857, 0.602): El coeficiente de variación indica que hay una relativa consistencia en la exploración a lo largo de las dimensiones, con la segunda dimensión mostrando más variabilidad relativa.

Curtosis (-1.097, -1.861, -1.053): La curtosis negativa indica que las posiciones visitadas tienen distribuciones más planas en comparación con una distribución normal, sugiriendo menos valores extremos.

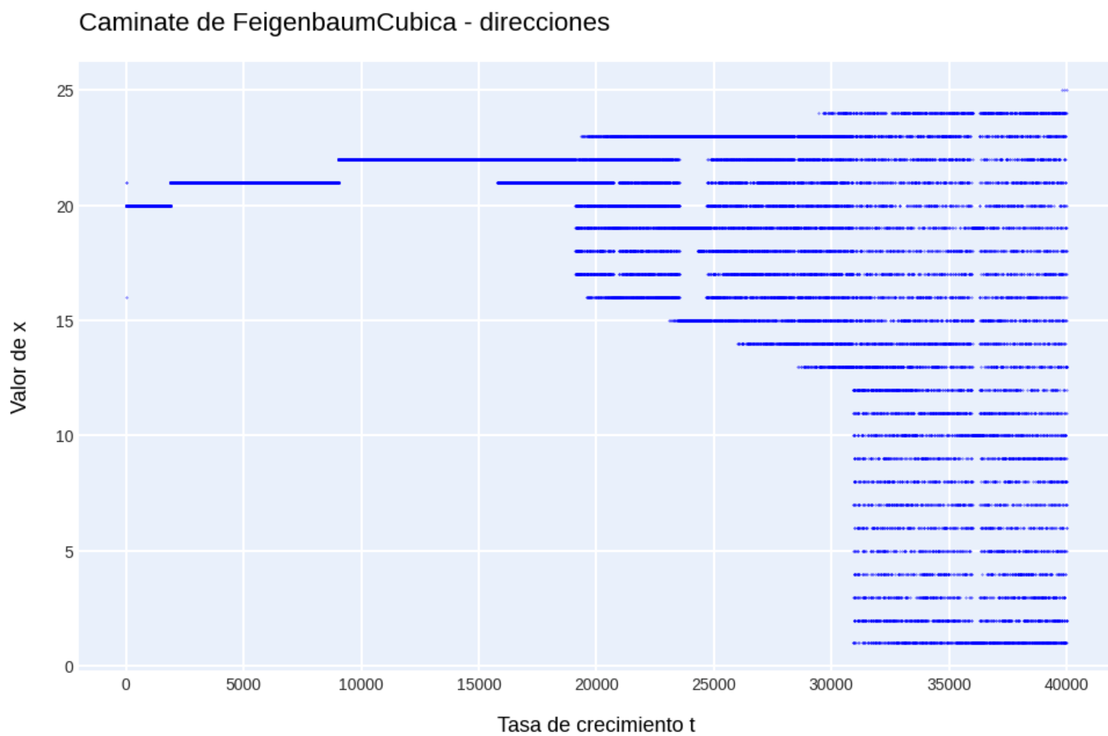
Entropía (2.684, 2.836, 2.953): La entropía indica un nivel moderado a alto de incertidumbre en la exploración de posiciones, lo que sugiere un balance entre aleatoriedad y repetición en el movimiento del caminante.

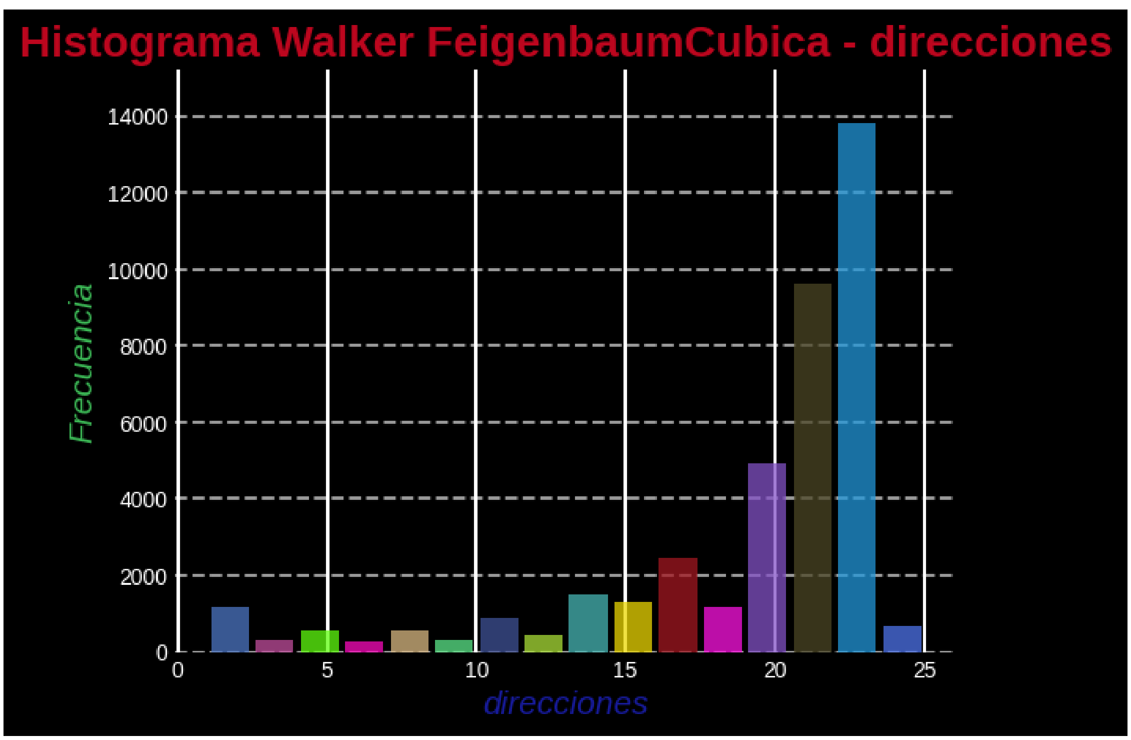
1.5 Caminante de Feigenbaum Cúbico



1.5.1 Direccion

[24]:
```
modelo = "feigenbaumcubica"
metrica = "direcciones"
folder = "FeigenbaumCubica"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
  ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
  ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```


Caminate de FeigenbaumCubica - direcciones
Valor de x
Tasa de crecimiento t
0
5
10
15
20
25
0
5000
10000
15000
20000
25000
30000
35000
40000


Histograma Walker FeigenbaumCubica - direcciones
Frecuencia
direcciones
14000
12000
10000
8000
6000
4000
2000
0
0
5
10
15
20
25

Interpretación de las Métricas Media (18.791): La media indica que, en promedio, las direcciones seleccionadas por el caminante están alrededor del índice 18.791.

Mediana (21.0): La mediana sugiere que la mitad de las direcciones seleccionadas son menores o iguales a 21 y la otra mitad son mayores o iguales a 21.

Moda (22, 11134): La moda muestra que la dirección 22 es la más frecuente, seleccionada 11,134 veces, lo que indica una fuerte preferencia del caminante por moverse en esta dirección.

Desviación Estándar (5.212): La desviación estándar de 5.212 indica que las direcciones seleccionadas varían considerablemente alrededor de la media.

Varianza (27.16): La varianza de 27.16 confirma una alta dispersión en las direcciones seleccionadas.

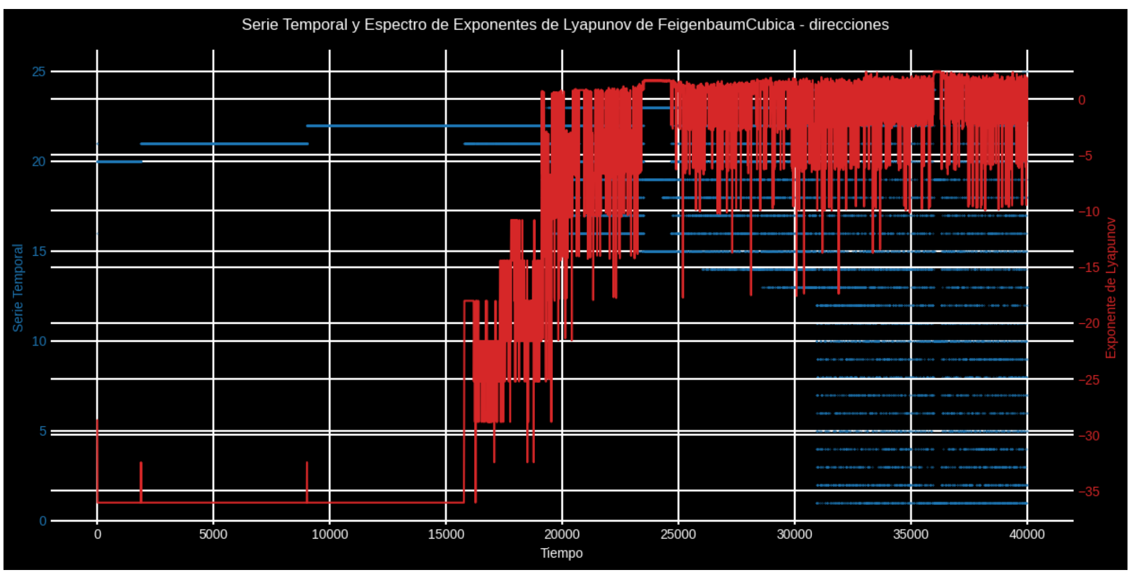
Asimetría (-1.944): La asimetría de -1.944 sugiere que la distribución está sesgada hacia la izquierda, indicando una cola más larga en el lado izquierdo de la distribución.

Curtosis (3.133): La curtosis de 3.133 indica que la distribución es más puntiaguda en comparación con una distribución normal, sugiriendo más valores extremos.

Entropía (2.384): La entropía de 2.384 indica un nivel moderado de incertidumbre o aleatoriedad en las direcciones seleccionadas por el caminante.

1.5.2 Exponentes de lyapounov y Kaplan Yorke

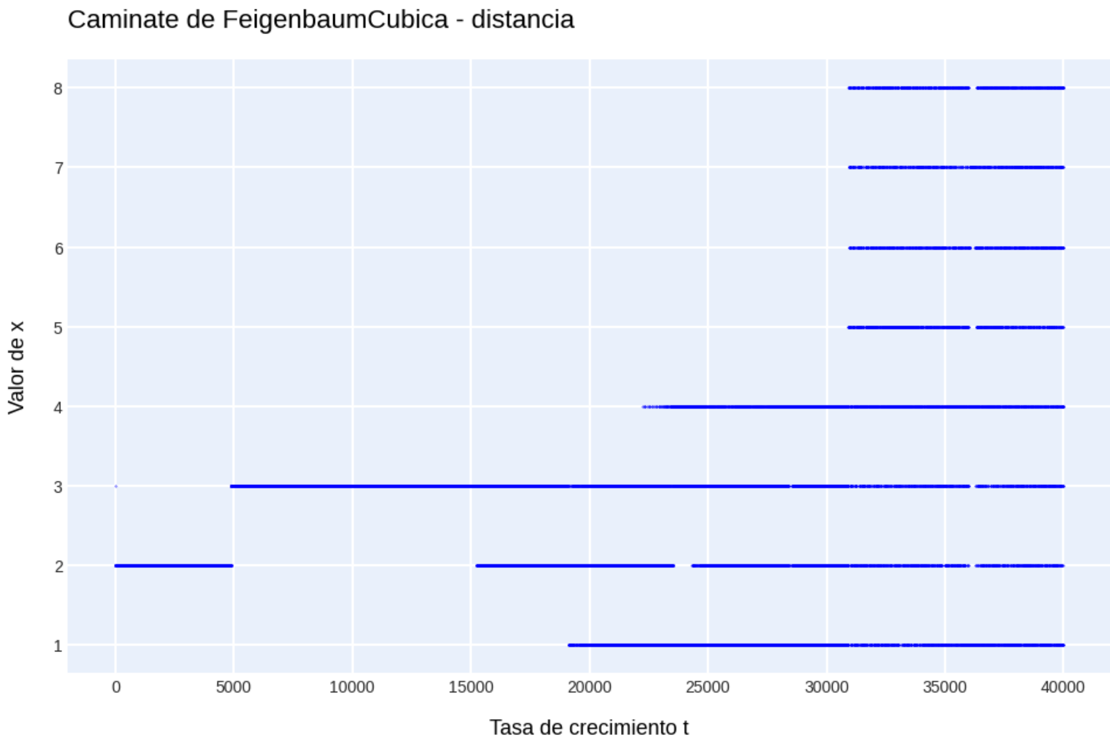
[25]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```

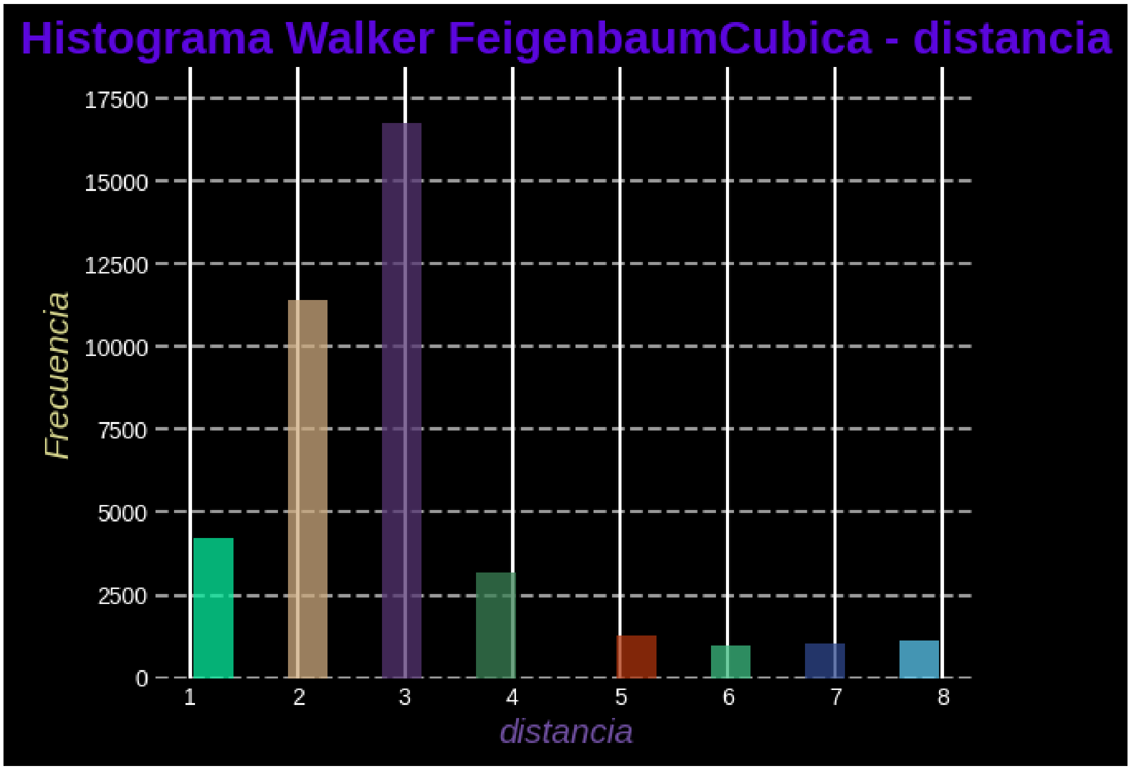



```
La dimension de Kaplan York es: 4.0
```

1.5.3 Distancia

[26]:
```
metrica = "distancia"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
    ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
    ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
res
```





[26]:

	Métrica	Valor
0	Media	2.965
1	Mediana	3.0
2	Moda	(3, 16780)
3	Media Geométrica	2.646
4	Rango	7
5	Desviación Estándar	1.498
6	Varianza	2.245
7	Asimetría	1.565
8	Coeficiente de Variación	0.505
9	Curtosis	2.838
10	Entropía	1.556
11	Percentil 25	2.0
12	Percentil 50	3.0
13	Percentil 75	3.0
14	Cuartil 1	2.0
15	Cuartil 2	3.0
16	Cuartil 3	3.0

Interpretación de las Métricas Media (2.965): La media indica que, en promedio, el caminante avanza aproximadamente 2.965 unidades por movimiento.

Mediana (3.0): La mediana sugiere que la mitad de las distancias recorridas son menores o iguales a 3 y la otra mitad son mayores o iguales a 3.

Moda (3, 16780): La moda muestra que la distancia de 3 unidades es la más frecuente, seleccionada 16,780 veces, lo que indica una fuerte preferencia por esta distancia.

Desviación Estándar (1.498): La desviación estándar de 1.498 sugiere una variabilidad moderada en las distancias recorridas.

Varianza (2.245): La varianza de 2.245 confirma la observación de la desviación estándar, indicando una dispersión moderada en las distancias recorridas.

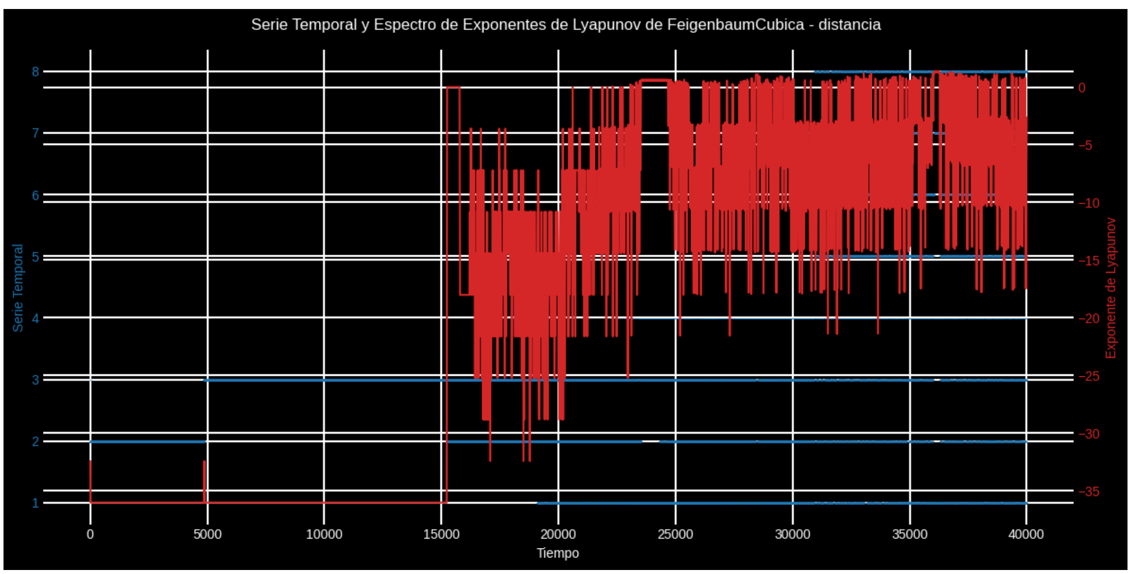
Asimetría (1.565): La asimetría de 1.565 sugiere que la distribución está sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (2.838): La curtosis de 2.838 indica que la distribución es más puntiaguda en comparación con una distribución normal, sugiriendo más valores extremos.

Entropía (1.556): La entropía de 1.556 indica un nivel moderado de incertidumbre en la selección de distancias.

1.5.4 Exponentes de lyapounov y Kaplan Yorke

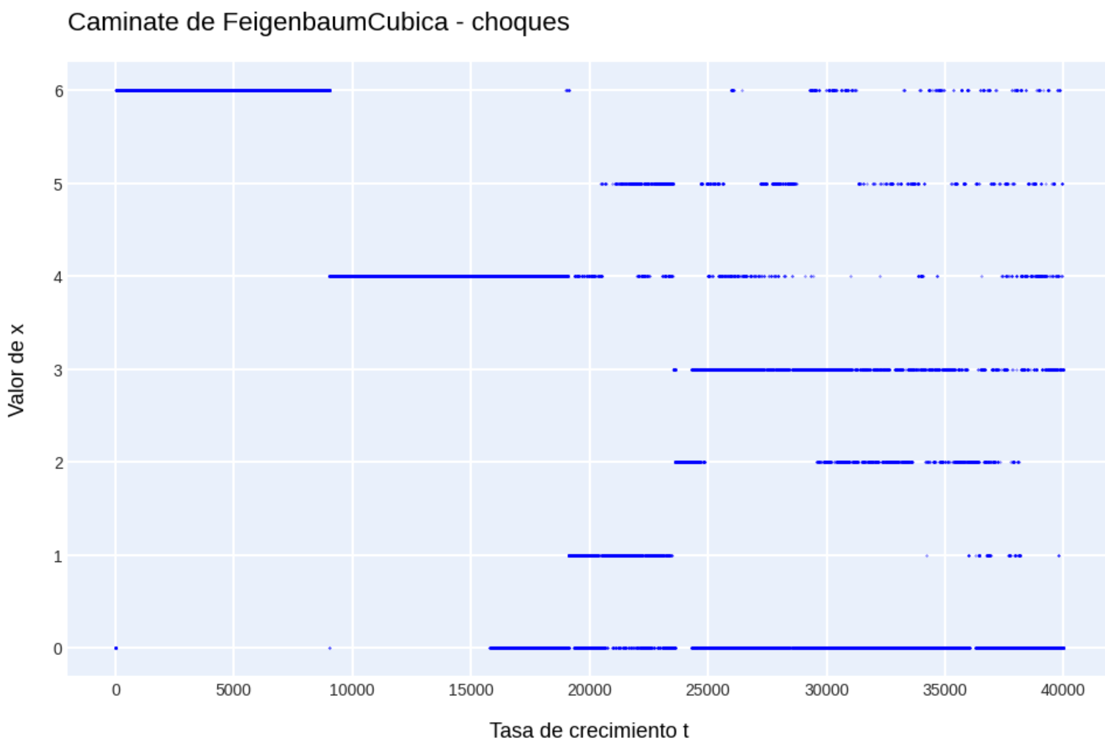
[27]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
    ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```

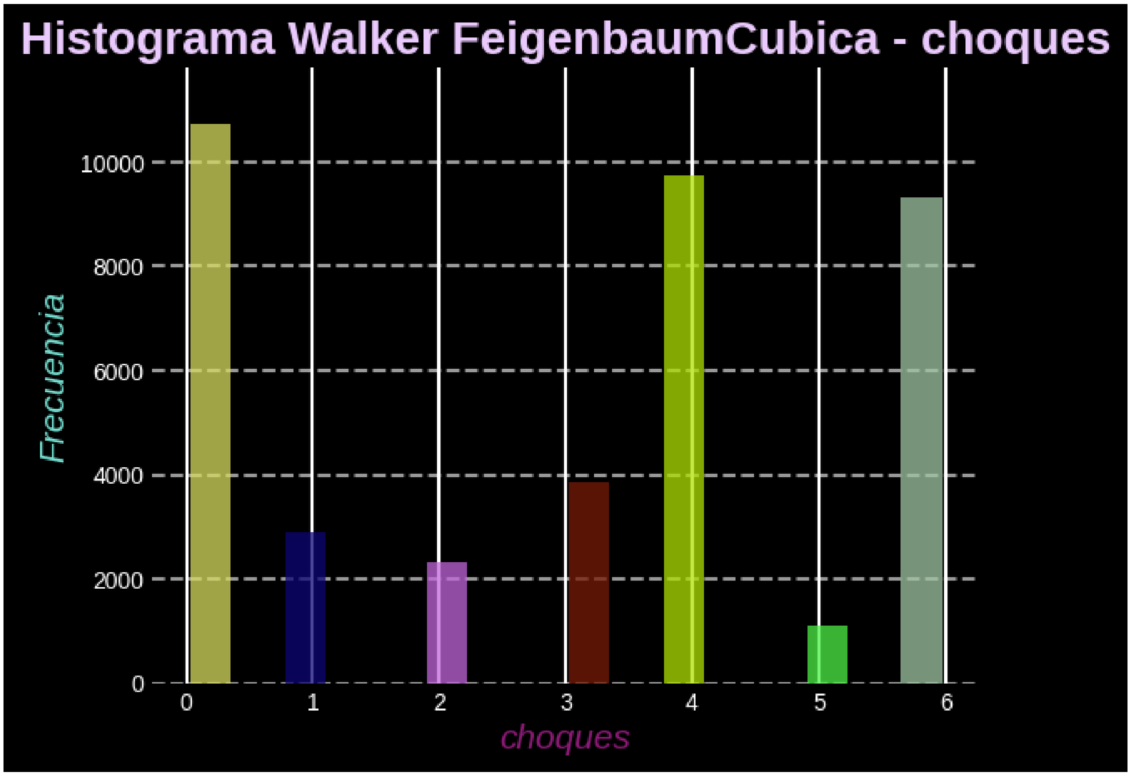



```
La dimension de Kaplan York es: 4.0
```

1.5.5 Choques

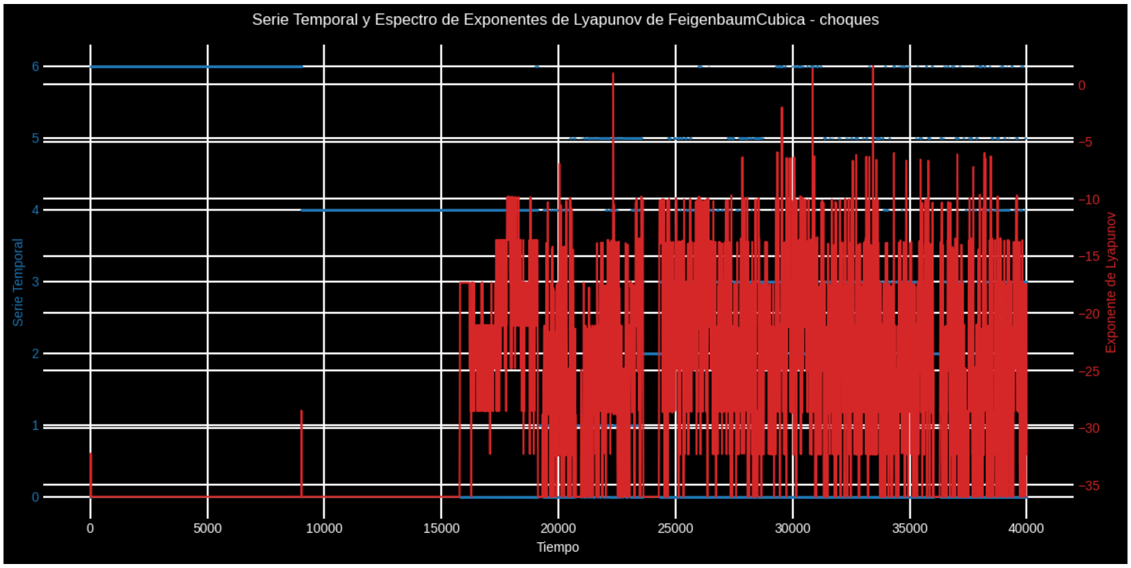
[28]:
```
metrica = "choques"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```





1.5.6 Exponentes de lyapounov y Kaplan Yorke

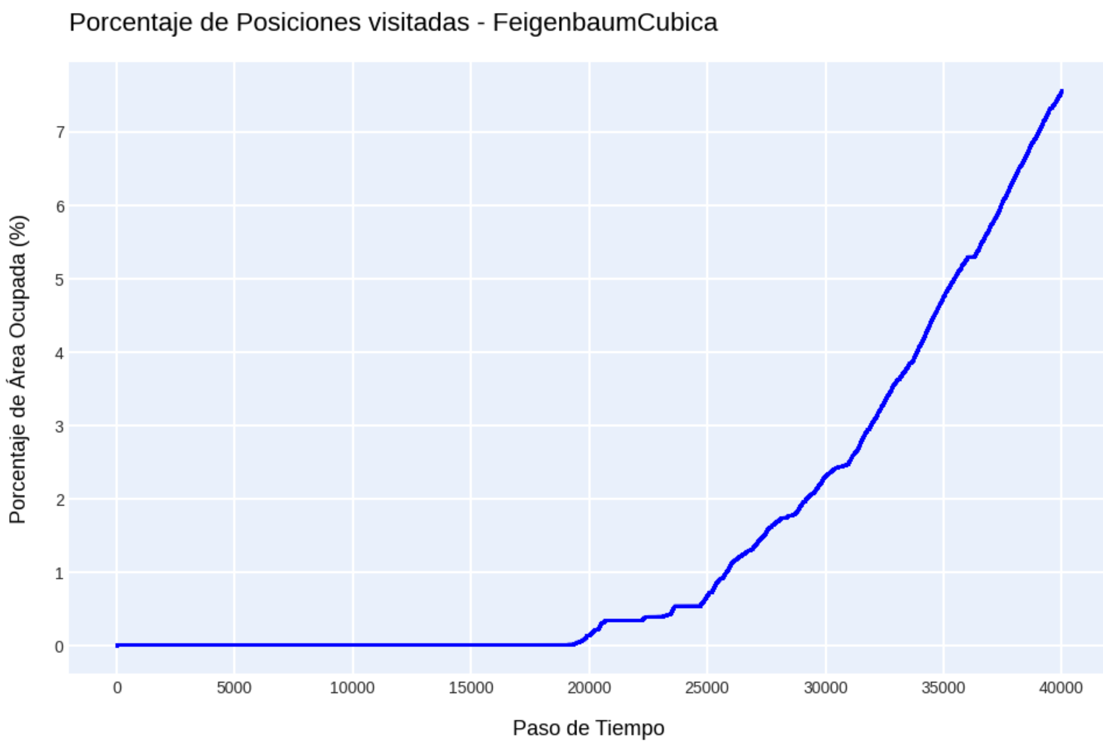
[29]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
    ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```




```
La dimension de Kaplan York es: 5.014936228212478
```

1.5.7 Posiciones

[30]:
```
metrica = "posiciones"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ";")

graficar_posicionesOcupadas(49, walker_feig, title=f"Porcentaje de Posiciones␣
 ↪visitadas - {folder}")
ex = DistribucionProbabilidadVector(np.array(walker_feig))
res = ex.calcular_metricas()
```



Interpretación de las Métricas Media (12.535, 20.445, 31.584): La media indica que, en promedio, las posiciones visitadas por el caminante están alrededor de estos valores en el espacio tridimensional, con un sesgo hacia ciertas áreas del cubo.

Mediana (6.0, 10.0, 43.0): La mediana sugiere que la mitad de las posiciones visitadas se encuentran por debajo de estos valores, indicando una concentración en ciertas áreas del cubo.

Moda ([0, 4, 49], [12610, 9435, 10182]): La moda muestra que ciertas posiciones específicas son visitadas con mucha más frecuencia que otras, lo que indica una fuerte preferencia o patrón repetitivo en el movimiento del caminante.

Desviación Estándar (17.186, 19.149, 18.739): La desviación estándar indica una considerable variabilidad en las posiciones visitadas, especialmente en la primera y segunda dimensiones.

Varianza (295.373, 366.701, 351.154): La varianza confirma la alta dispersión en las posiciones visitadas, con una mayor dispersión en la segunda dimensión.

Asimetría (1.266, 0.548, -0.687): La asimetría sugiere que la distribución de las posiciones visitadas tiene diferentes sesgos en cada dimensión, con una distribución más equilibrada en la segunda dimensión y distribuciones sesgadas hacia valores mayores en las otras dos dimensiones.

Coeficiente de Variación (1.371, 0.937, 0.593): El coeficiente de variación indica que hay una relativa consistencia en la exploración a lo largo de las dimensiones, con la primera dimensión mostrando más variabilidad relativa.

Curtosis (-0.068, -1.495, -1.235): La curtosis negativa indica que las posiciones visitadas tienen distribuciones más planas en comparación con una distribución normal, sugiriendo menos valores extremos.

Entropía (2.569, 2.561, 2.819): La entropía indica un nivel moderado a alto de incertidumbre en la exploración de posiciones, lo que sugiere un balance entre aleatoriedad y repetición en el movimiento del caminante.

1.5.8 Interpretación de las Métricas de Caos para Feigenbaum Cúbico

Serie Temporal y Espectro de Exponentes de Lyapunov - Direcciones La gráfica muestra la evolución temporal de las direcciones seleccionadas por el caminante aleatorio, superpuesta con el espectro de exponentes de Lyapunov.

- **Serie Temporal**: Se observa una tendencia de las direcciones alrededor de valores específicos (mayores a 15), con un comportamiento más errático hacia el final de la serie.
- **Exponentes de Lyapunov**: Los exponentes de Lyapunov negativos indican estabilidad local y sensibilidad a las condiciones iniciales. La variabilidad en estos valores a lo largo del tiempo refleja la naturaleza caótica del sistema.
- **Dimensión de Kaplan Yorke**: 4.0. Este valor sugiere una complejidad en la dinámica del sistema, con una alta dimensión fractal que es típica en sistemas caóticos.

Serie Temporal y Espectro de Exponentes de Lyapunov - Distancia Esta gráfica ilustra la evolución de las distancias recorridas en cada paso por el caminante aleatorio, junto con el espectro de exponentes de Lyapunov.

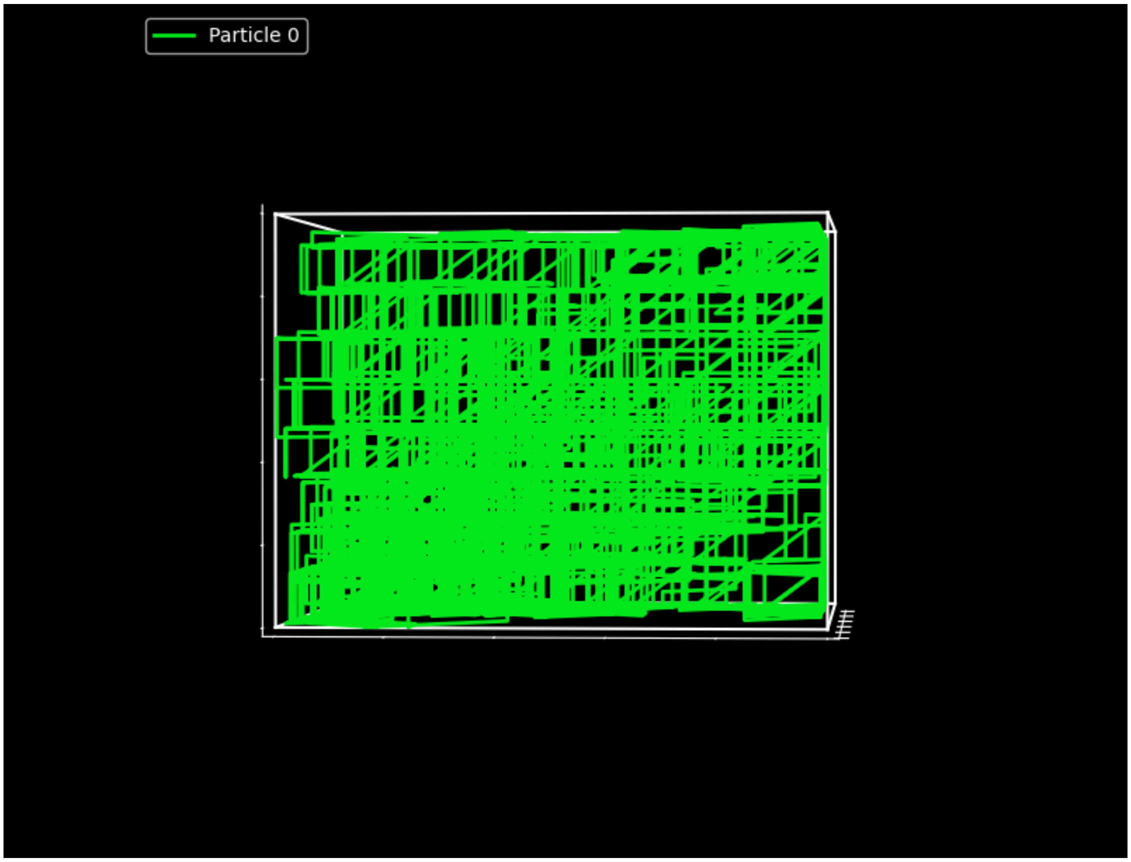
- **Serie Temporal**: Las distancias se concentran en valores bajos (1 a 4), lo que sugiere que el caminante realiza pequeños desplazamientos con más frecuencia.
- **Exponentes de Lyapunov**: Al igual que en la gráfica de direcciones, los exponentes negativos predominan, indicando regiones de estabilidad y comportamiento caótico.
- **Dimensión de Kaplan Yorke**: 4.0. Este valor refuerza la idea de una dinámica compleja y caótica en el movimiento del caminante.

Serie Temporal y Espectro de Exponentes de Lyapunov - Choques La gráfica muestra el comportamiento de los choques del caminante con las paredes del cubo, en combinación con los exponentes de Lyapunov.

- **Serie Temporal**: La mayoría de los datos indican choques con la pared etiquetada como '0' (sin choque), con algunos episodios de choques con otras paredes, especialmente hacia el final del tiempo.
- **Exponentes de Lyapunov**: Los exponentes negativos y su variabilidad reflejan el carácter caótico del sistema y su sensibilidad a las condiciones iniciales.
- **Dimensión de Kaplan Yorke**: 4.0. Este resultado indica un nivel muy alto de complejidad y caos en el sistema, lo que podría estar relacionado con la alta frecuencia y variabilidad de los choques.

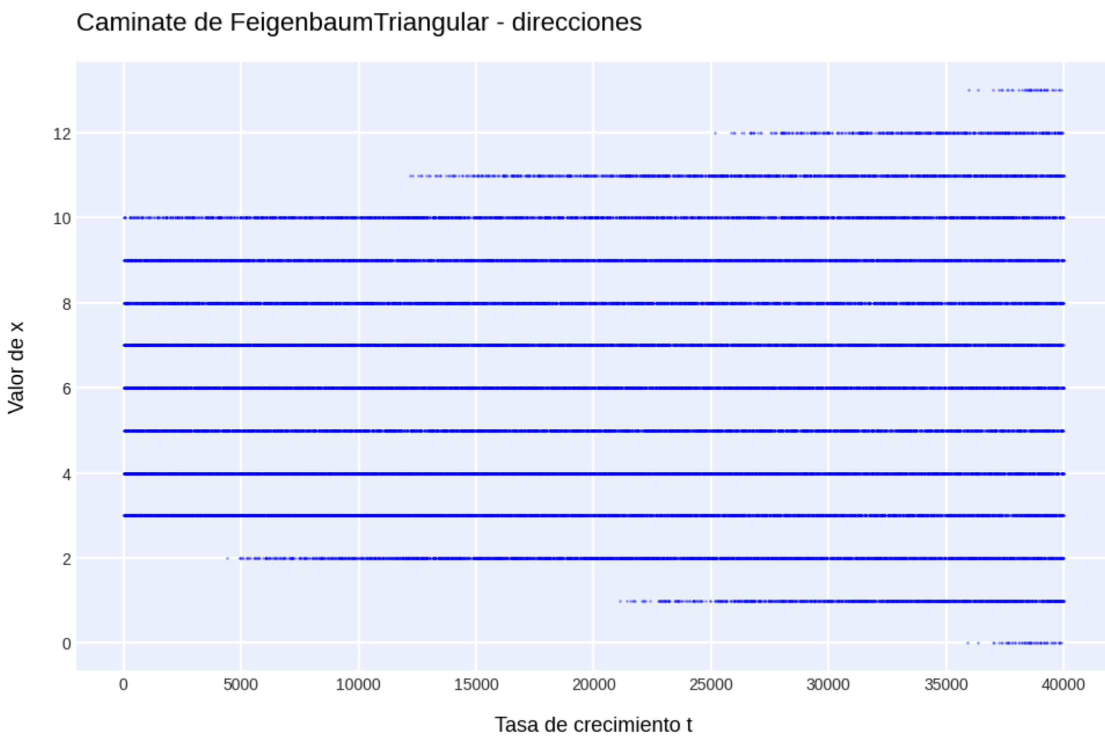
Estas métricas y gráficas confirman que el comportamiento del caminante aleatorio, alimentado por modelos caóticos, exhibe las características típicas de sistemas caóticos: sensibilidad a las condiciones iniciales, comportamiento impredecible y una estructura fractal compleja.

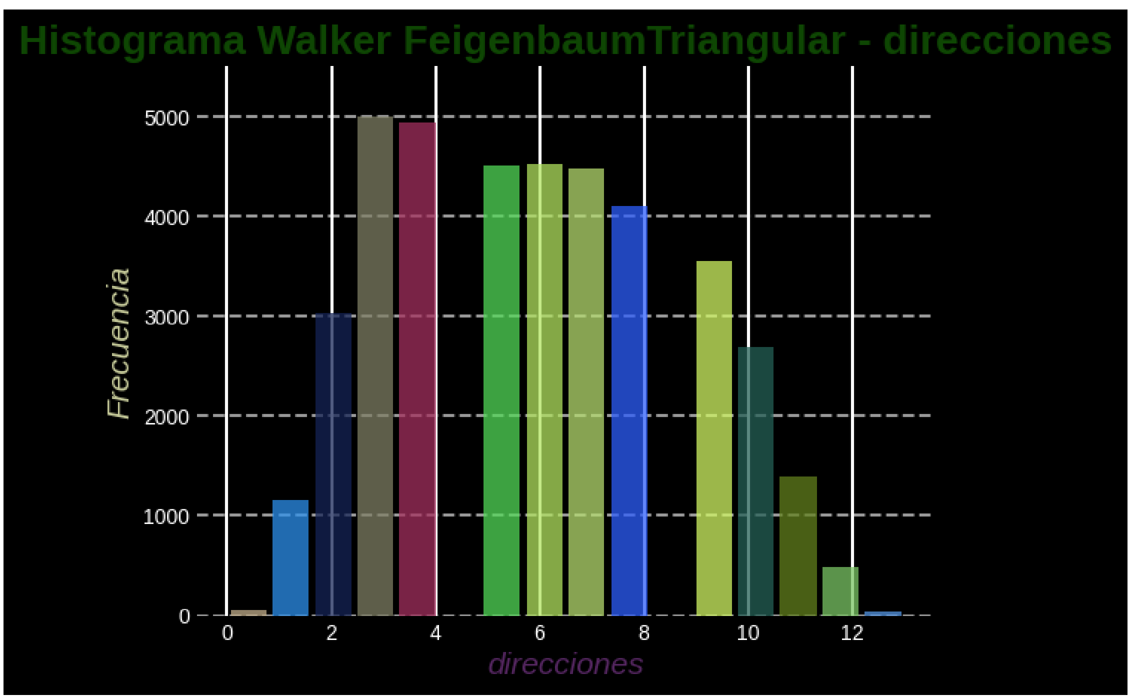
1.6 Caminante de Feigenbaum Triangular



1.6.1 Dirección

[31]:
```
modelo = "feigenbaumtriangular"
metrica = "direcciones"
folder = "FeigenbaumTriangular"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```


Caminate de FeigenbaumTriangular - direcciones
Valor de x
Tasa de crecimiento t
0
2
4
6
8
10
12
0
5000
10000
15000
20000
25000
30000
35000
40000


Histograma Walker FeigenbaumTriangular - direcciones
Frecuencia
direcciones
0
1000
2000
3000
4000
5000
0
2
4
6
8
10
12

Interpretación de las Métricas Media (5.918): La media indica que, en promedio, las direcciones seleccionadas por el caminante están alrededor del índice 5.918.

Mediana (6.0): La mediana sugiere que la mitad de las direcciones seleccionadas son menores o iguales a 6 y la otra mitad son mayores o iguales a 6.

Moda (3, 5008): La moda muestra que la dirección 3 es la más frecuente, seleccionada 5,008 veces, lo que indica una fuerte preferencia del caminante por moverse en esta dirección.

Desviación Estándar (2.736): La desviación estándar de 2.736 indica que las direcciones seleccionadas varían considerablemente alrededor de la media.

Varianza (7.483): La varianza de 7.483 confirma una alta dispersión en las direcciones seleccionadas.

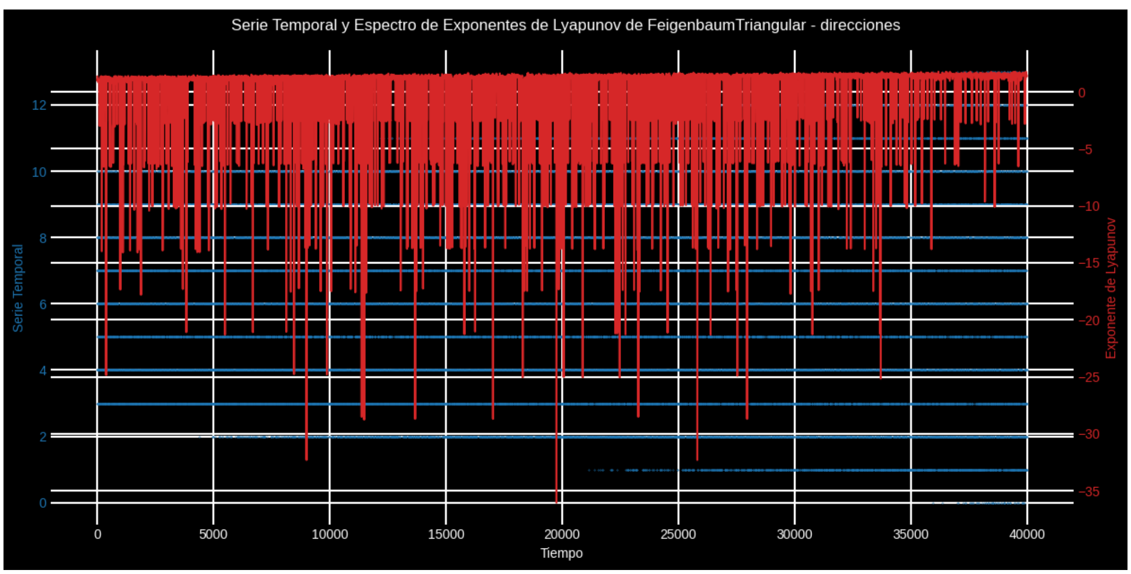
Asimetría (0.177): La asimetría cercana a 0 sugiere que la distribución de las direcciones es bastante simétrica.

Curtosis (-0.875): La curtosis de -0.875 indica que la distribución es más plana en comparación con una distribución normal, sugiriendo menos valores extremos.

Entropía (2.373): La entropía de 2.373 indica un nivel moderado de incertidumbre o aleatoriedad en las direcciones seleccionadas por el caminante.

1.6.2 EXPONENTES DE LYAPOUNOV Y KAPLAN YORKE

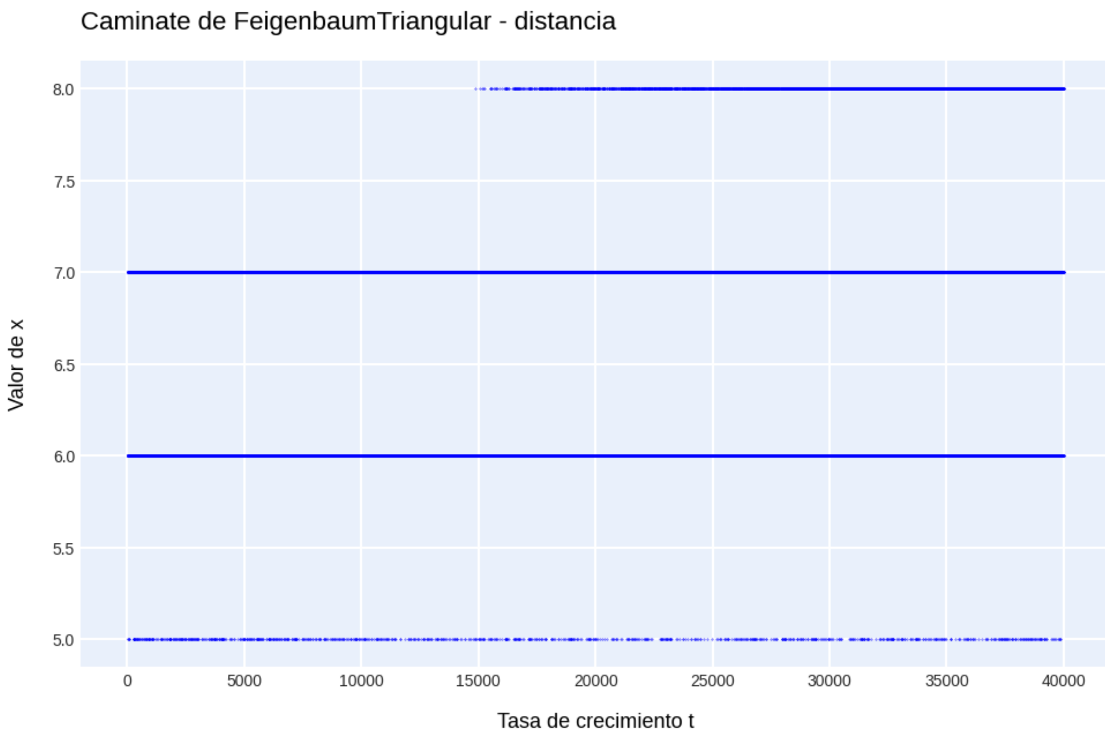
[32]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```

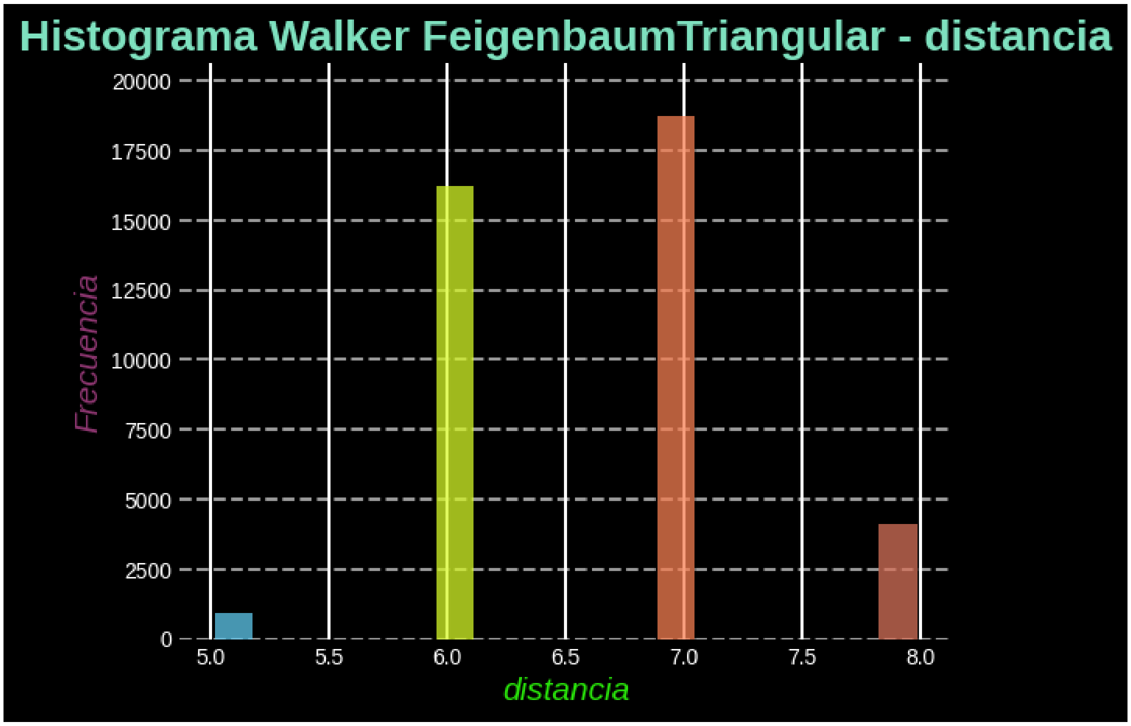



```
La dimension de Kaplan York es: 4.002957255766532
```

1.6.3 Distancia

[33]:
```
metrica = "distancia"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
 ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
 ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```





Interpretación de las Métricas Media (6.65): La media indica que, en promedio, el caminante avanza aproximadamente 6.65 unidades por movimiento.

Mediana (7.0): La mediana sugiere que la mitad de las distancias recorridas son menores o iguales a 7 y la otra mitad son mayores o iguales a 7.

Moda (7, 18754): La moda muestra que la distancia de 7 unidades es la más frecuente, seleccionada 18,754 veces, lo que indica una fuerte preferencia por esta distancia.

Desviación Estándar (0.692): La desviación estándar de 0.692 sugiere una baja variabilidad en las distancias recorridas.

Varianza (0.478): La varianza de 0.478 confirma la observación de la desviación estándar, indicando una baja dispersión en las distancias recorridas.

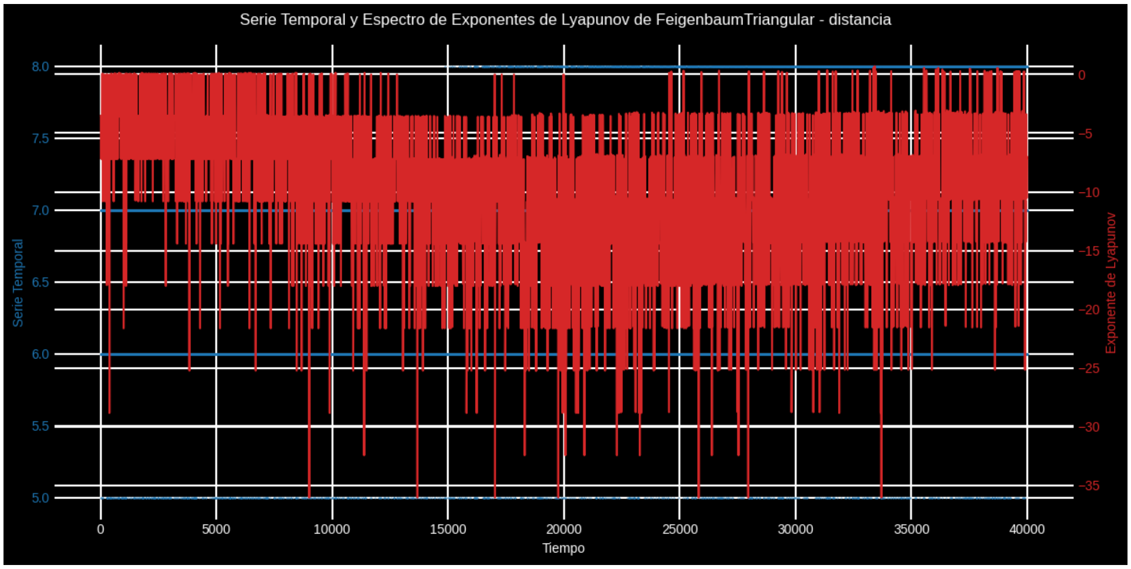
Asimetría (0.169): La asimetría cercana a 0 indica que la distribución de las distancias es bastante simétrica.

Curtosis (-0.418): La curtosis de -0.418 indica que la distribución es más plana en comparación con una distribución normal, sugiriendo menos valores extremos.

Entropía (1.042): La entropía de 1.042 indica un nivel bajo de incertidumbre en la selección de distancias.

1.6.4 Exponentes de lyapounov y Kaplan Yorke

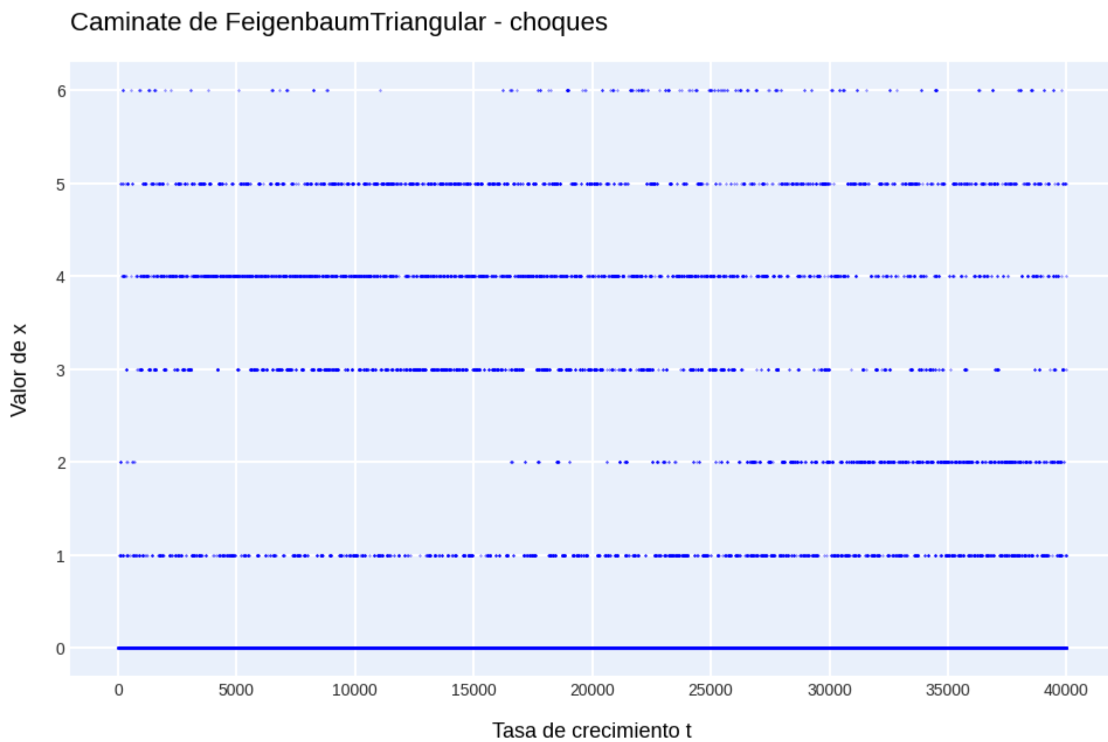
[34]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```

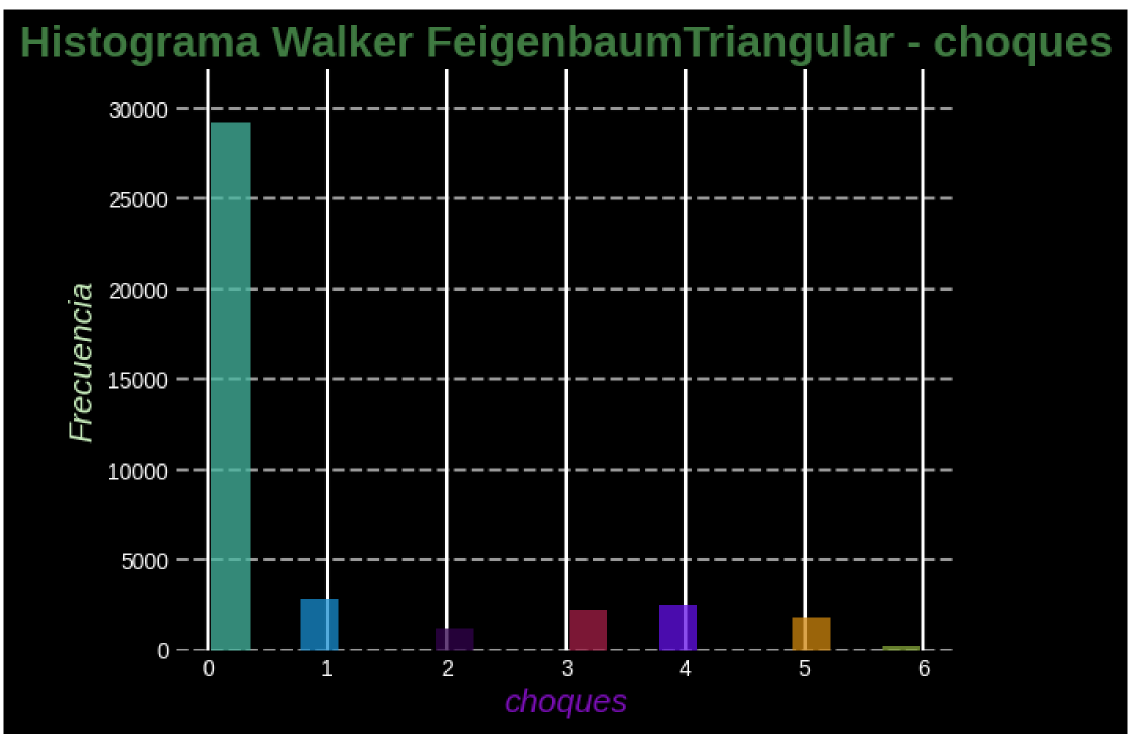



```
La dimension de Kaplan York es: 4.25
```

1.6.5 Choques

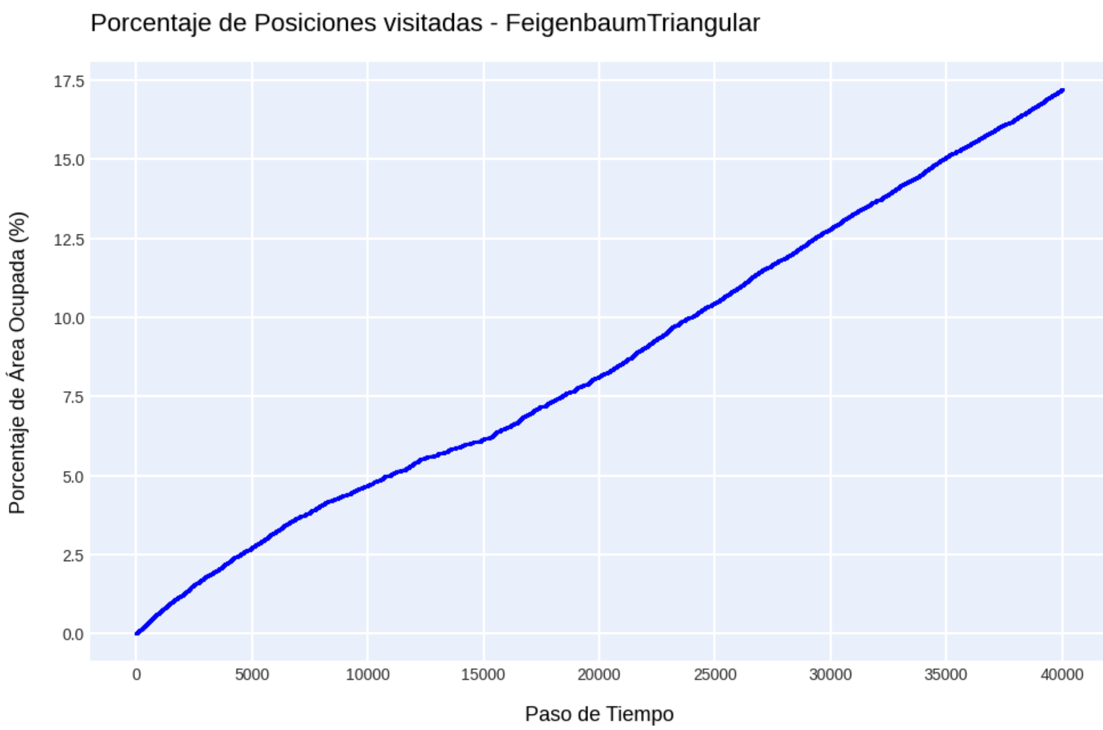
[35]:
```
metrica = "choques"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```


Caminate de FeigenbaumTriangular - choques
Valor de x
Tasa de crecimiento t
0
1
2
3
4
5
6
0
5000
10000
15000
20000
25000
30000
35000
40000


Histograma Walker FeigenbaumTriangular - choques
Frecuencia
choques
0
5000
10000
15000
20000
25000
30000
0
1
2
3
4
5
6

1.6.6 Posiciones

[36]:
```
metrica = "posiciones"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ";")

graficar_posicionesOcupadas(49, walker_feig, title=f"Porcentaje de Posiciones␣
 ↪visitadas - {folder}")
ex = DistribucionProbabilidadVector(np.array(walker_feig))
res = ex.calcular_metricas()
```



Interpretación de las Métricas Media (0.805): La media indica que, en promedio, el caminante choca con las paredes del cubo aproximadamente 0.805 veces por movimiento.

Mediana (0.0): La mediana sugiere que la mitad de los movimientos no resultan en choques.

Moda (0, 29257): La moda muestra que no chocar es el evento más frecuente, ocurriendo 29,257 veces.

Desviación Estándar (1.539): La desviación estándar de 1.539 sugiere una variabilidad considerable en el número de choques.

Varianza (2.367): La varianza de 2.367 confirma una alta dispersión en el número de choques.

Asimetría (1.756): La asimetría de 1.756 sugiere que la distribución está sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (1.669): La curtosis de 1.669 indica que la distribución es más puntiaguda en comparación con una distribución normal, sugiriendo más valores extremos.

Entropía (1.022): La entropía de 1.022 indica un nivel bajo de incertidumbre en la ocurrencia de choques.

1.6.7 Serie Temporal y Espectro de Exponentes de Lyapunov - FeigenbaumTriangular (direcciones)

La gráfica muestra la evolución temporal de las direcciones seleccionadas por el caminante aleatorio, superpuesta con el espectro de exponentes de Lyapunov.

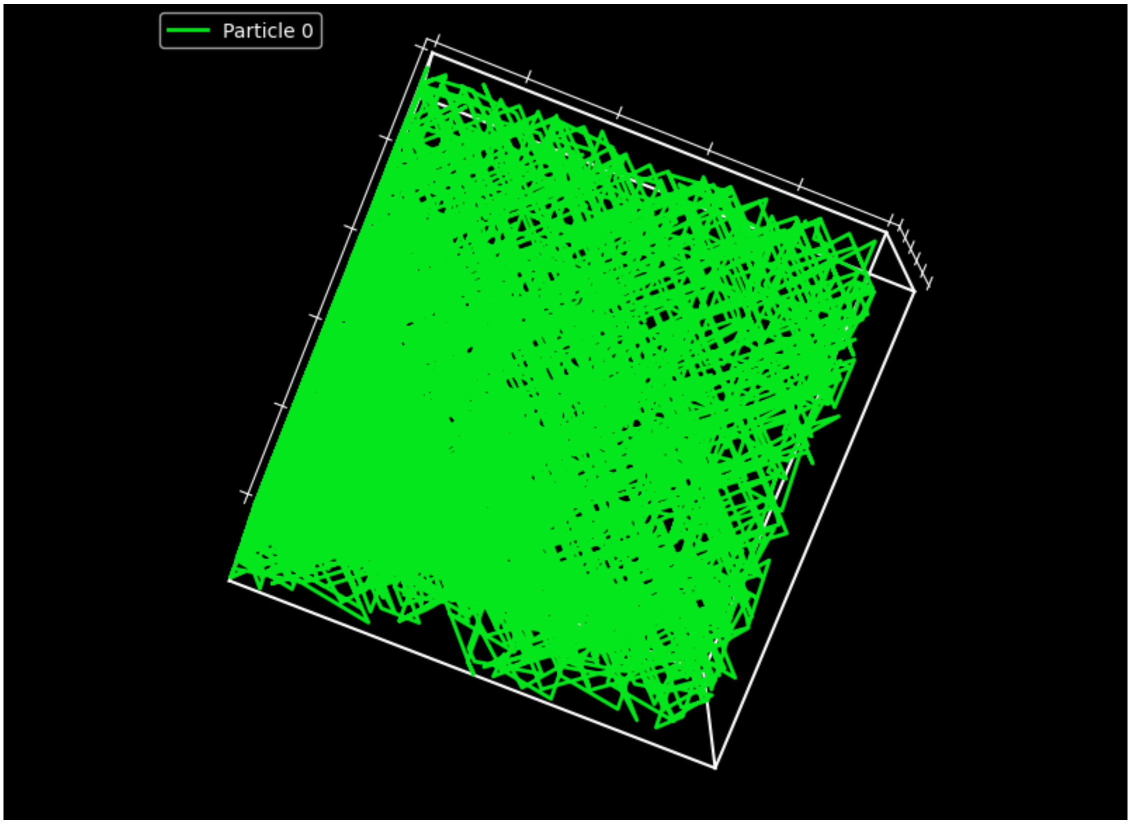
- **Serie Temporal**: Las direcciones se concentran en un rango amplio (de 0 a más de 10), lo que indica una mayor diversidad en las direcciones tomadas por el caminante. Hay episodios de alta variabilidad a lo largo del tiempo.
- **Exponentes de Lyapunov**: Los exponentes de Lyapunov negativos predominan, lo que sugiere que el sistema es localmente estable en muchas regiones, pero con una sensibilidad a las condiciones iniciales que induce comportamiento caótico.
- **Dimensión de Kaplan Yorke**: 4.002957255766532. Este valor indica una complejidad moderada en la dinámica del sistema, con una estructura fractal típica de sistemas caóticos.

1.6.8 Serie Temporal y Espectro de Exponentes de Lyapunov - FeigenbaumTriangular (distancia)

Esta gráfica ilustra la evolución de las distancias recorridas en cada paso por el caminante aleatorio, junto con el espectro de exponentes de Lyapunov.

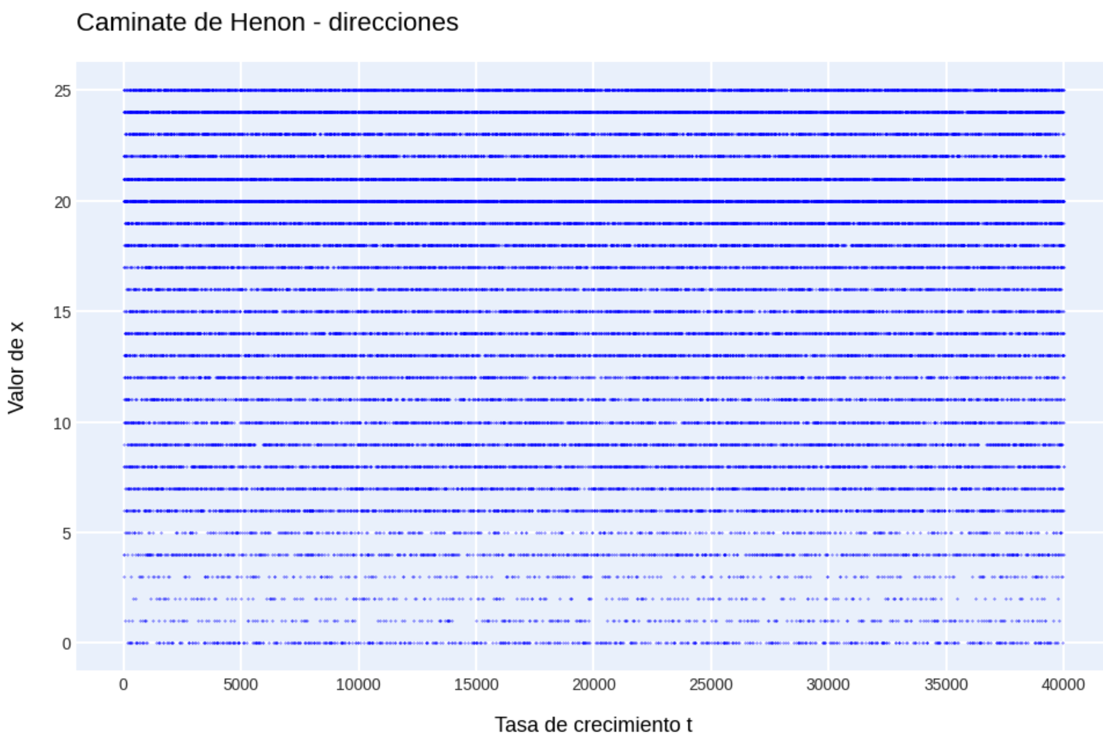
- **Serie Temporal**: Las distancias se concentran principalmente en valores bajos (alrededor de 5 a 8), lo que sugiere que el caminante realiza desplazamientos pequeños y medianos con más frecuencia.
- **Exponentes de Lyapunov**: Al igual que en la gráfica de direcciones, los exponentes de Lyapunov negativos predominan, lo que indica regiones de estabilidad local y comportamiento caótico.
- **Dimensión de Kaplan Yorke**: 4.25. Este valor refuerza la idea de una dinámica compleja y caótica en el movimiento del caminante, con una estructura fractal.

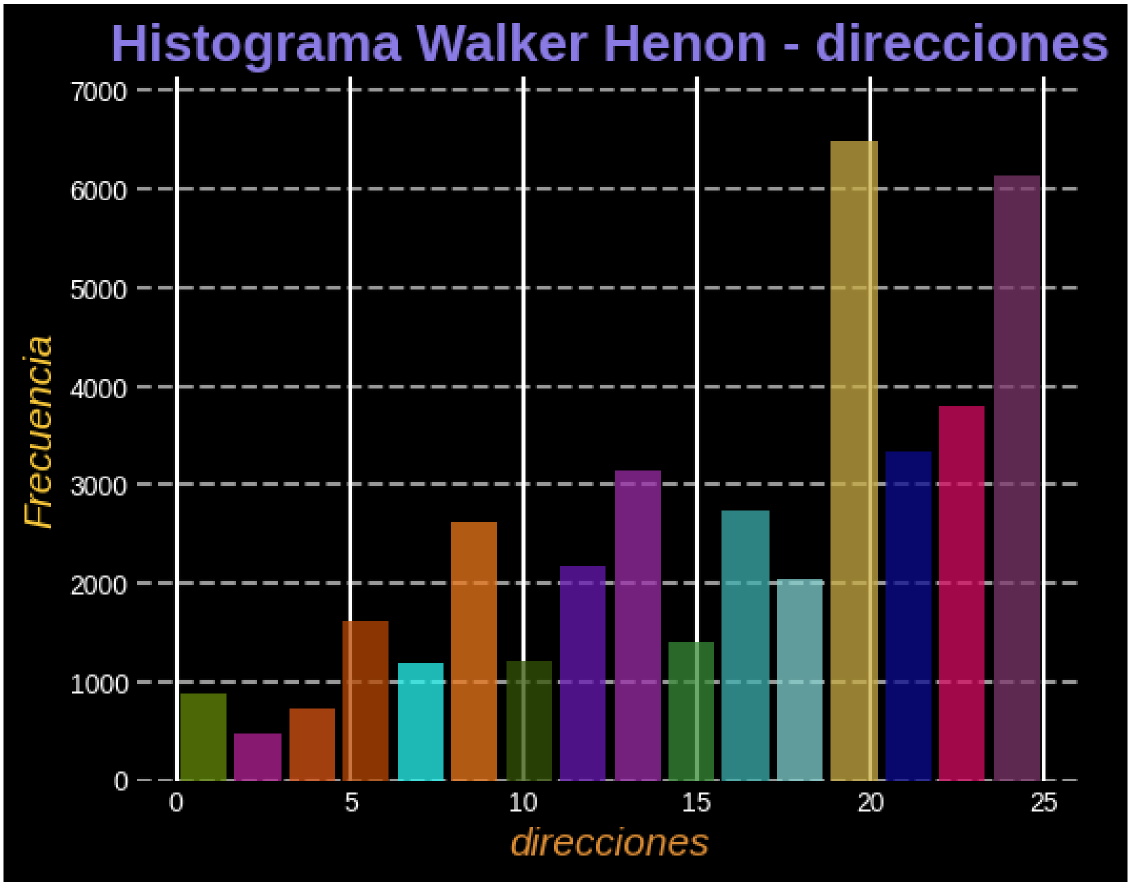
1.7 Atractor de Henón



1.7.1 Direcciones

[37]:
```
modelo = "transfhenon"
metrica = "direcciones"
folder = "Henon"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
  ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
  ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```


Caminate de Henon - direcciones
Valor de x
Tasa de crecimiento t
25
20
15
10
5
0
0
5000
10000
15000
20000
25000
30000
35000
40000



Interpretación de las Métricas Media (16.507): La media indica que, en promedio, las direcciones seleccionadas por el caminante están alrededor del índice 16.507.

Mediana (18.0): La mediana sugiere que la mitad de las direcciones seleccionadas son menores o iguales a 18 y la otra mitad son mayores o iguales a 18.

Moda (20, 4333): La moda muestra que la dirección 20 es la más frecuente, seleccionada 4,333 veces, lo que indica una fuerte preferencia del caminante por moverse en esta dirección.

Desviación Estándar (6.525): La desviación estándar de 6.525 indica que las direcciones seleccionadas varían considerablemente alrededor de la media.

Varianza (42.58): La varianza de 42.58 confirma una alta dispersión en las direcciones seleccionadas.

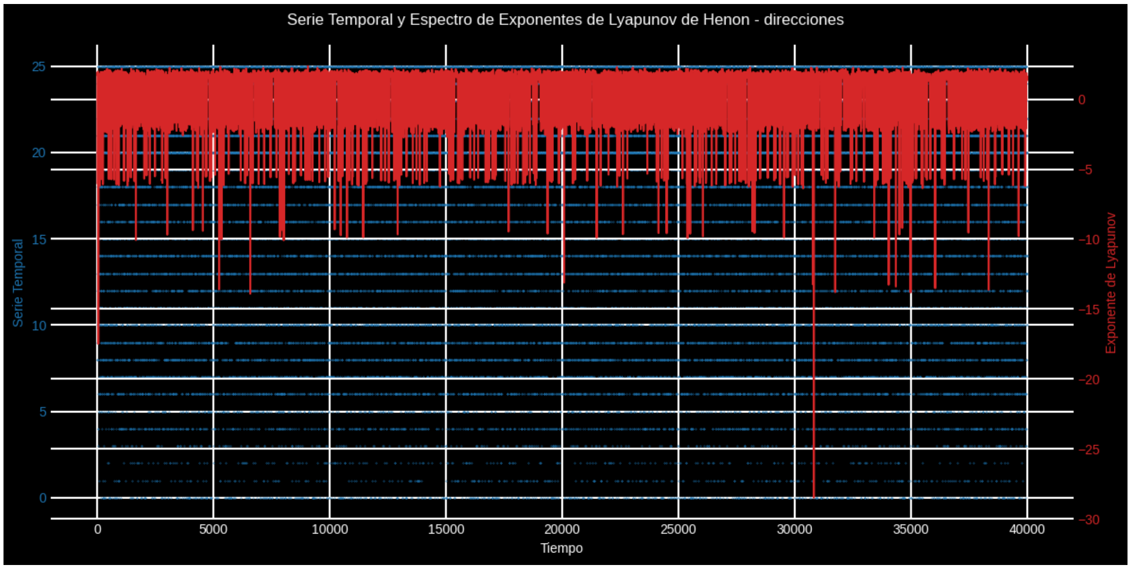
Asimetría (-0.644): La asimetría de -0.644 sugiere que la distribución está sesgada hacia la izquierda, indicando una cola más larga en el lado izquierdo de la distribución.

Curtosis (-0.583): La curtosis de -0.583 indica que la distribución es más plana en comparación con una distribución normal, sugiriendo menos valores extremos.

Entropía (3.056): La entropía de 3.056 indica un alto nivel de incertidumbre o aleatoriedad en las direcciones seleccionadas por el caminante.

1.7.2 Exponentes de lyapounov y Kaplan Yorke

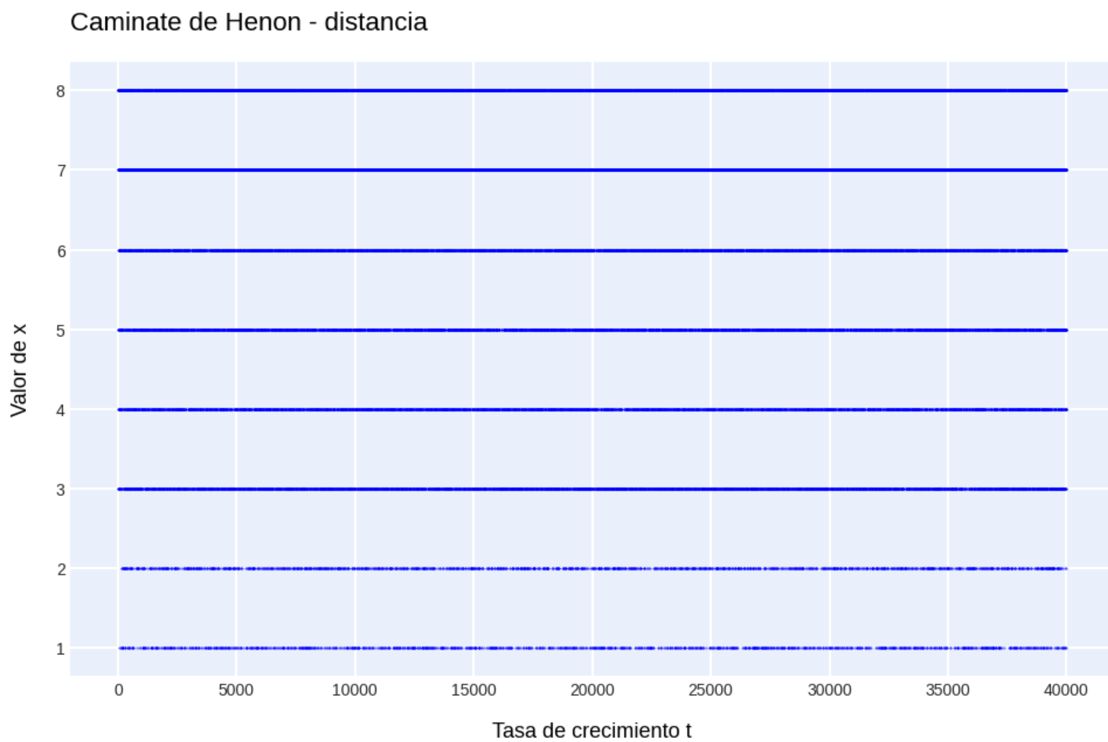
[38]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```



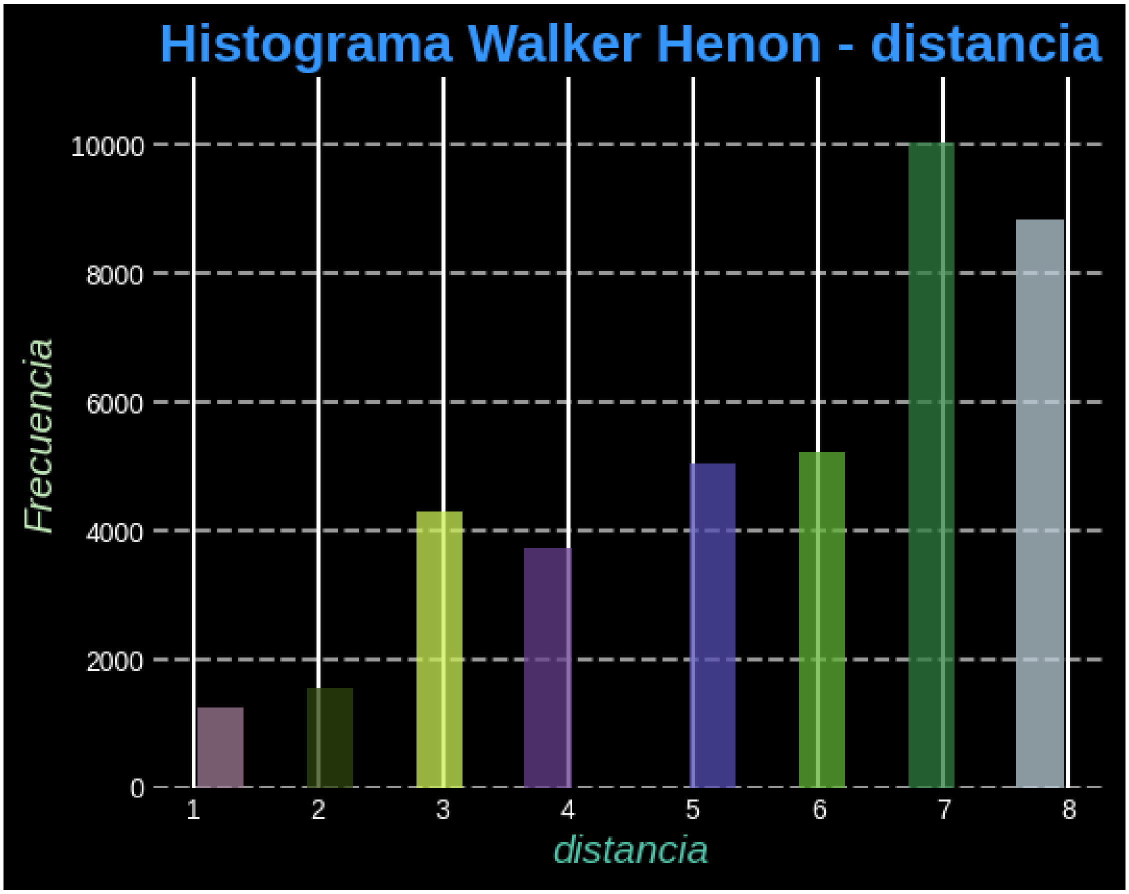

```
La dimension de Kaplan York es: 4.033920214989325
```

1.7.3 Distancias

[39]:
```
metrica = "distancia"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```



Caminate de Henon - distancia
Valor de x
Tasa de crecimiento t
1
2
3
4
5
6
7
8
0
5000
10000
15000
20000
25000
30000
35000
40000



Interpretación de las Métricas Media (5.747): La media indica que, en promedio, el caminante avanza aproximadamente 5.747 unidades por movimiento.

Mediana (6.0): La mediana sugiere que la mitad de las distancias recorridas son menores o iguales a 6 y la otra mitad son mayores o iguales a 6.

Moda (7, 10052): La moda muestra que la distancia de 7 unidades es la más frecuente, seleccionada 10,052 veces, lo que indica una fuerte preferencia por esta distancia.

Desviación Estándar (1.985): La desviación estándar de 1.985 sugiere una variabilidad moderada en las distancias recorridas.

Varianza (3.94): La varianza de 3.94 confirma la observación de la desviación estándar, indicando una dispersión moderada en las distancias recorridas.

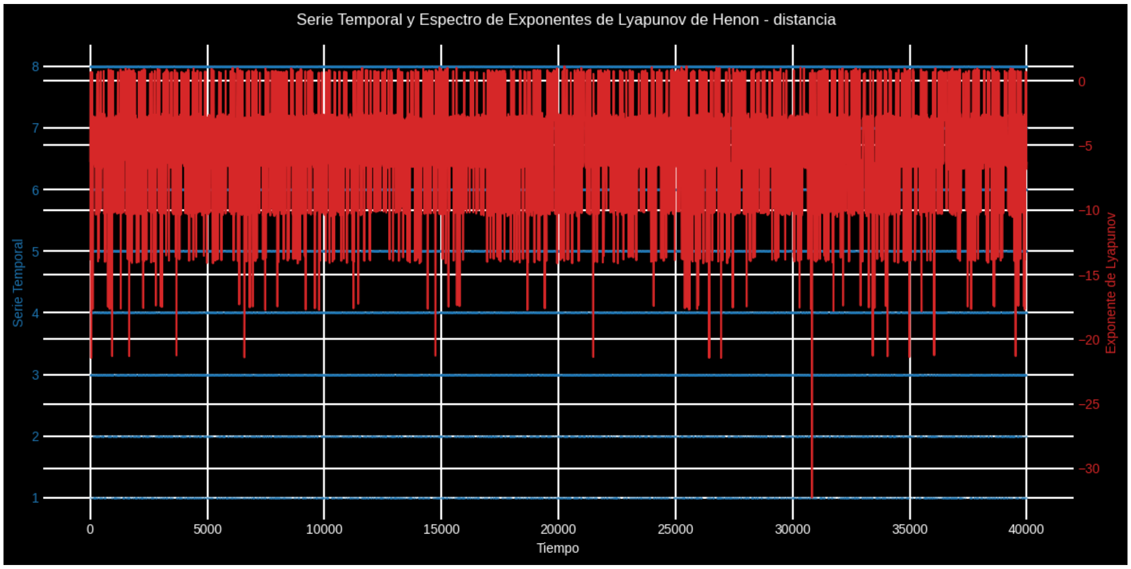
Asimetría (-0.656): La asimetría de -0.656 sugiere que la distribución está sesgada hacia la izquierda, indicando una cola más larga en el lado izquierdo de la distribución.

Curtosis (-0.628): La curtosis de -0.628 indica que la distribución es más plana en comparación con una distribución normal, sugiriendo menos valores extremos.

Entropía (1.903): La entropía de 1.903 indica un nivel moderado de incertidumbre en la selección de distancias.

1.7.4 Exponentes de lyapounov y Kaplan Yorke

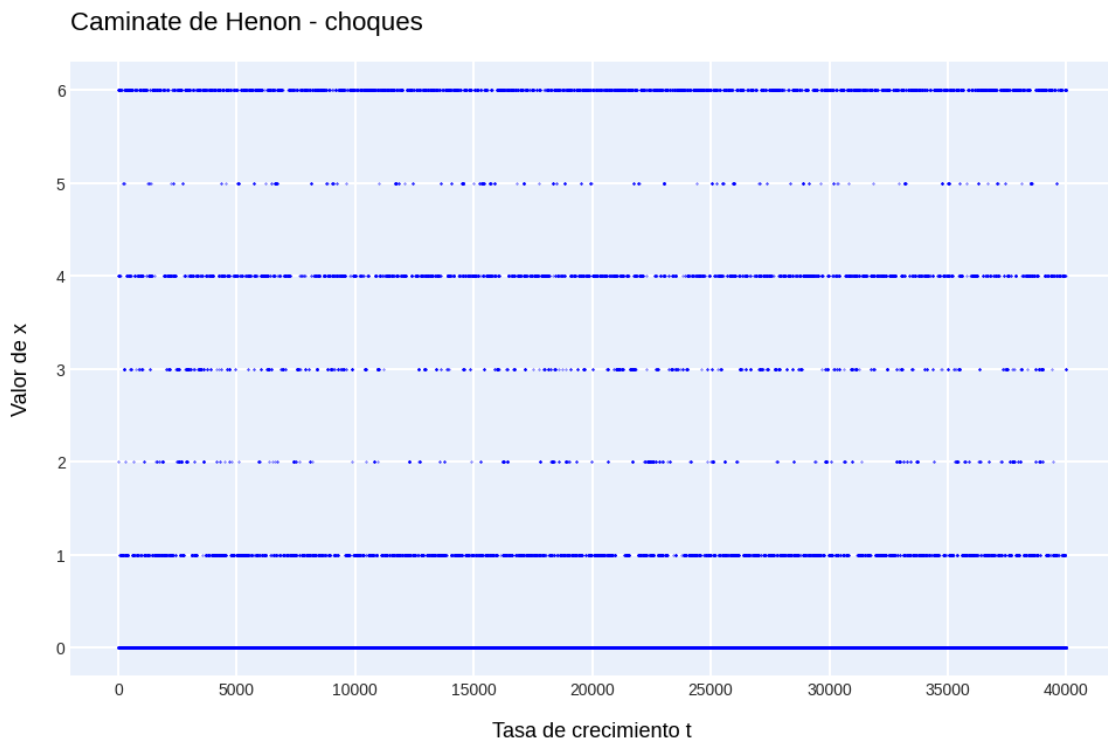
[40]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```

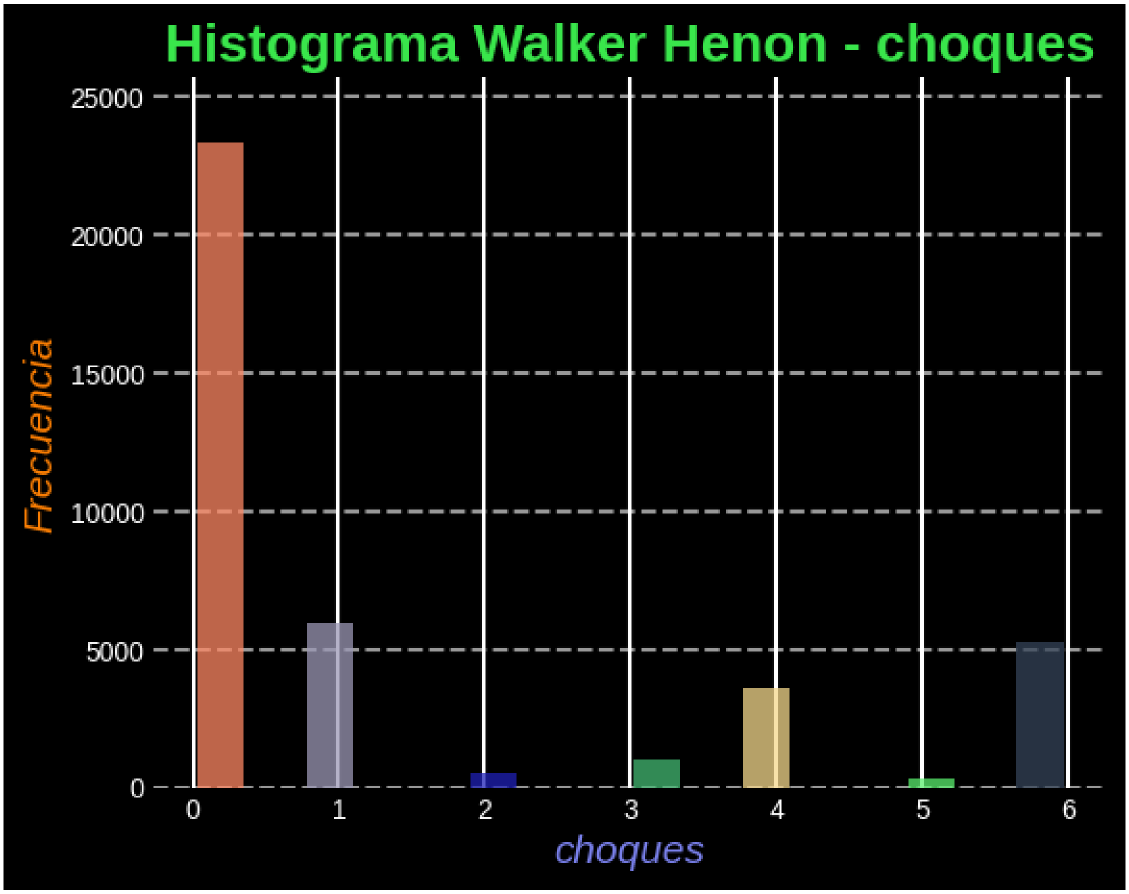



```
La dimension de Kaplan York es: 4.138906299937377
```

1.7.5 Choques

[41]:
```
metrica = "choques"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```


Caminate de Henon - choques
Valor de x
Tasa de crecimiento t
0
1
2
3
4
5
6
0
5000
10000
15000
20000
25000
30000
35000
40000



Interpretación de las Métricas Media (1.44): La media indica que, en promedio, el caminante choca con las paredes del cubo aproximadamente 1.44 veces por movimiento.

Mediana (0.0): La mediana sugiere que la mitad de los movimientos no resultan en choques.

Moda (0, 23362): La moda muestra que no chocar es el evento más frecuente, ocurriendo 23,362 veces.

Desviación Estándar (2.178): La desviación estándar de 2.178 sugiere una variabilidad considerable en el número de choques.

Varianza (4.745): La varianza de 4.745 confirma una alta dispersión en el número de choques.

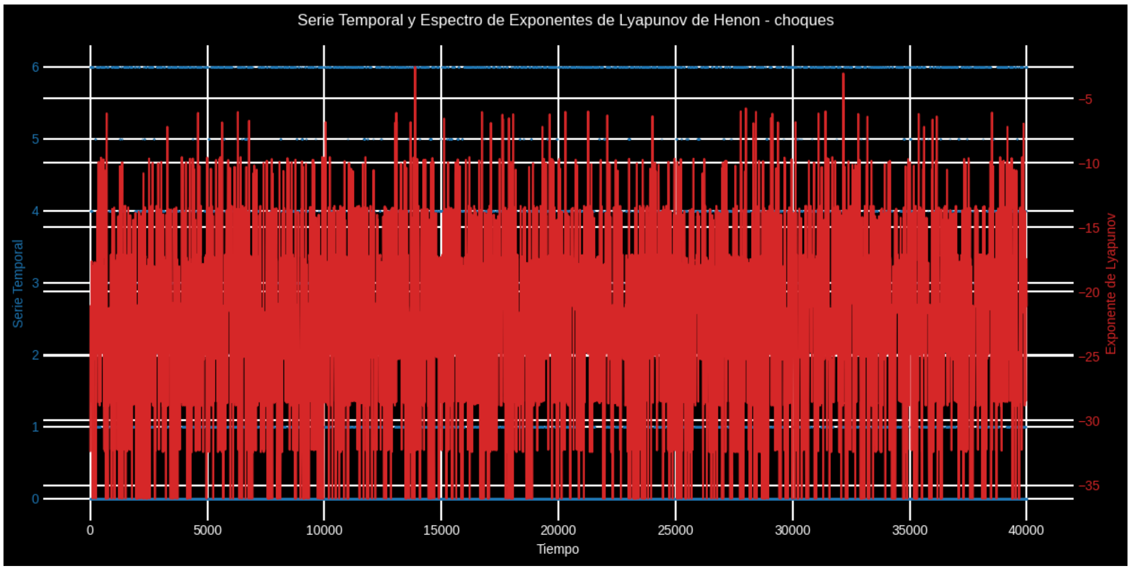
Asimetría (1.233): La asimetría de 1.233 sugiere que la distribución está sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (-0.113): La curtosis de -0.113 indica que la distribución es más plana en comparación con una distribución normal, sugiriendo menos valores extremos.

Entropía (1.267): La entropía de 1.267 indica un nivel moderado de incertidumbre en la ocurrencia de choques.

1.7.6 Exponentes de lyapounov y Kaplan Yorke

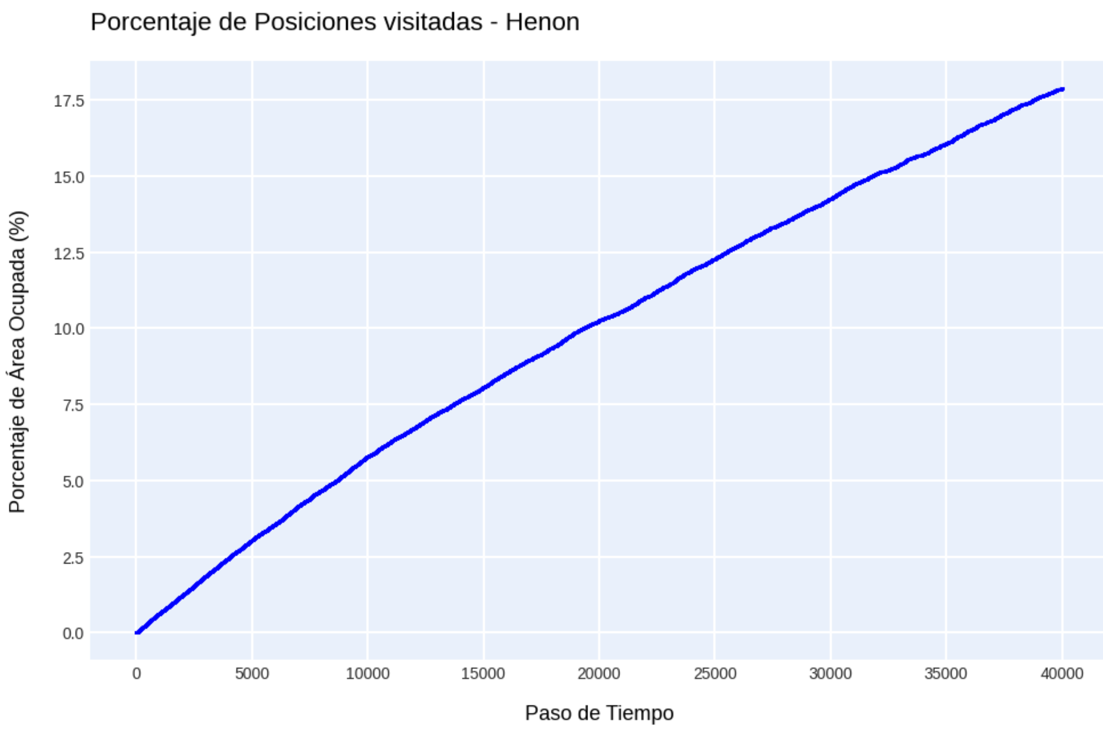
[42]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
 ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```




```
La dimension de Kaplan York es: 1.0260431731713948
```

1.7.7 Posiciones

[43]:
```
metrica = "posiciones"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ";")

graficar_posicionesOcupadas(49, walker_feig, title=f"Porcentaje de Posiciones␣
 ↪visitadas - {folder}")
ex = DistribucionProbabilidadVector(np.array(walker_feig))
res = ex.calcular_metricas()
```



1.7.8 Serie Temporal y Espectro de Exponentes de Lyapunov - Henon (direcciones)

- **Serie Temporal**:
  - **Descripción**: La gráfica muestra la evolución de las direcciones seleccionadas por el caminante aleatorio a lo largo del tiempo.
  - **Observaciones**: Las direcciones se concentran en un rango de aproximadamente 10 a 25, lo que indica una gran diversidad en las direcciones tomadas por el caminante. Hay episodios de alta variabilidad.
  - **Interpretación**: Esto sugiere que el caminante aleatorio tiene un comportamiento impredecible y variado en términos de direcciones.
- **Exponentes de Lyapunov**:
  - **Descripción**: Los exponentes de Lyapunov están superpuestos a la serie temporal de direcciones.
  - **Observaciones**: Predominan los exponentes negativos.
  - **Interpretación**: La presencia predominante de exponentes negativos indica que el sistema es localmente estable en muchas regiones, pero aún así presenta sensibilidad a las condiciones iniciales, lo cual es indicativo de comportamiento caótico.
- **Dimensión de Kaplan-Yorke**:
  - **Valor**: 4.03390214989325
  - **Interpretación**: Este valor indica una complejidad moderada en la dinámica del sistema, con una estructura fractal típica de sistemas caóticos.

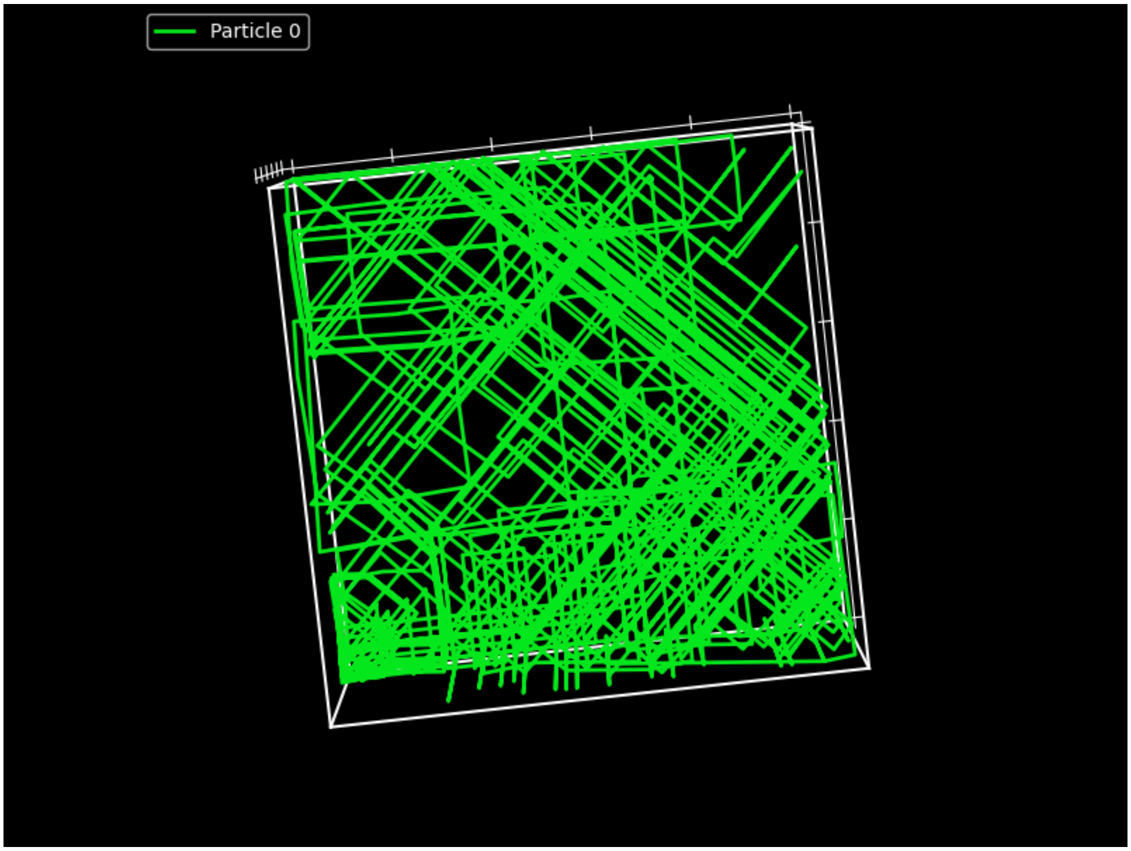
1.7.9 Serie Temporal y Espectro de Exponentes de Lyapunov - Henon (distancia)

- **Serie Temporal**:
  - **Descripción**: La gráfica muestra la evolución de las distancias recorridas en cada paso por el caminante aleatorio a lo largo del tiempo.
  - **Observaciones**: Las distancias se concentran principalmente en valores entre 5 y 8.
  - **Interpretación**: Esto sugiere que el caminante realiza desplazamientos pequeños y medianos con más frecuencia.
- **Exponentes de Lyapunov**:
  - **Descripción**: Los exponentes de Lyapunov están superpuestos a la serie temporal de distancias.
  - **Observaciones**: Predominan los exponentes negativos.
  - **Interpretación**: La presencia de exponentes negativos refuerza la idea de regiones de estabilidad local, aunque con comportamiento caótico.
- **Dimensión de Kaplan-Yorke**:
  - **Valor**: 4.138906299937377
  - **Interpretación**: Este valor sugiere una dinámica compleja y caótica en el movimiento del caminante, con una estructura fractal.

1.7.10 Serie Temporal y Espectro de Exponentes de Lyapunov - Henon (choques)

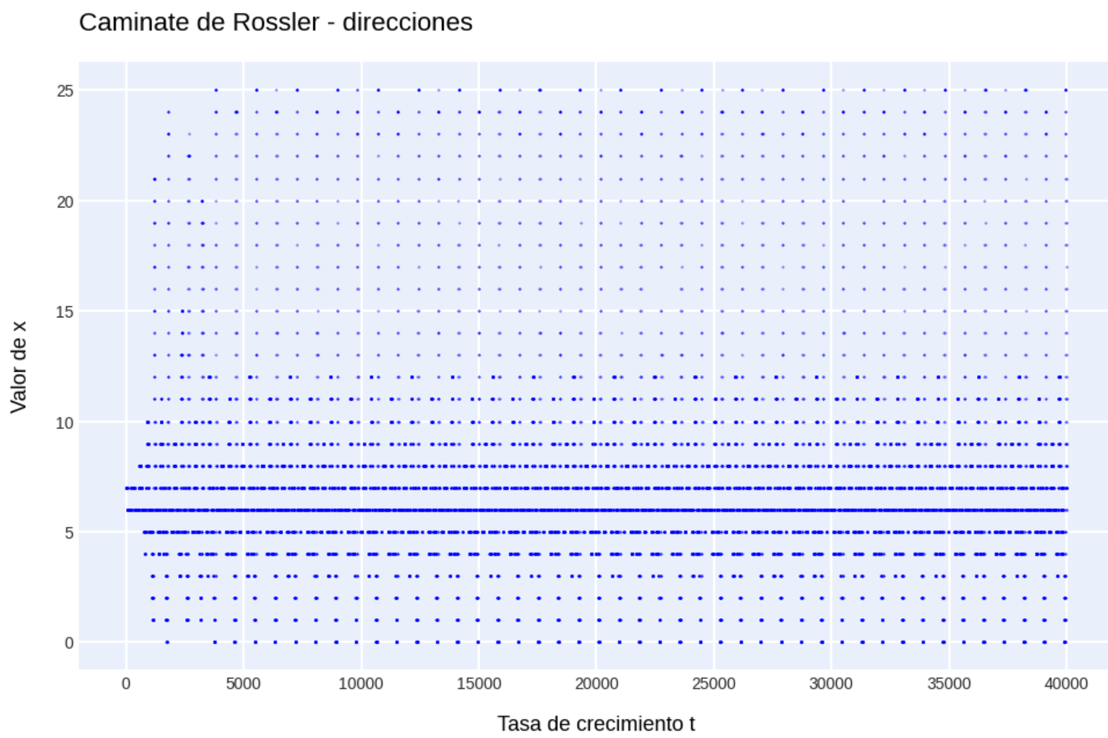
- **Serie Temporal**:
  - **Descripción**: La gráfica muestra la evolución de los choques (colisiones) del caminante aleatorio a lo largo del tiempo.
  - **Observaciones**: Los choques se concentran en un rango de aproximadamente 0 a 6, con una alta variabilidad.
  - **Interpretación**: Esto indica un comportamiento impredecible y variado en términos de choques.
- **Exponentes de Lyapunov**:
  - **Descripción**: Los exponentes de Lyapunov están superpuestos a la serie temporal de choques.
  - **Observaciones**: Predominan los exponentes negativos.
  - **Interpretación**: La presencia predominante de exponentes negativos indica estabilidad local en muchas regiones, aunque con comportamiento caótico.
- **Dimensión de Kaplan-Yorke**:
  - **Valor**: 1.0260431731713948
  - **Interpretación**: Este valor indica una menor complejidad en la dinámica del sistema comparado con las gráficas anteriores, aunque todavía presenta una estructura fractal y comportamiento caótico.

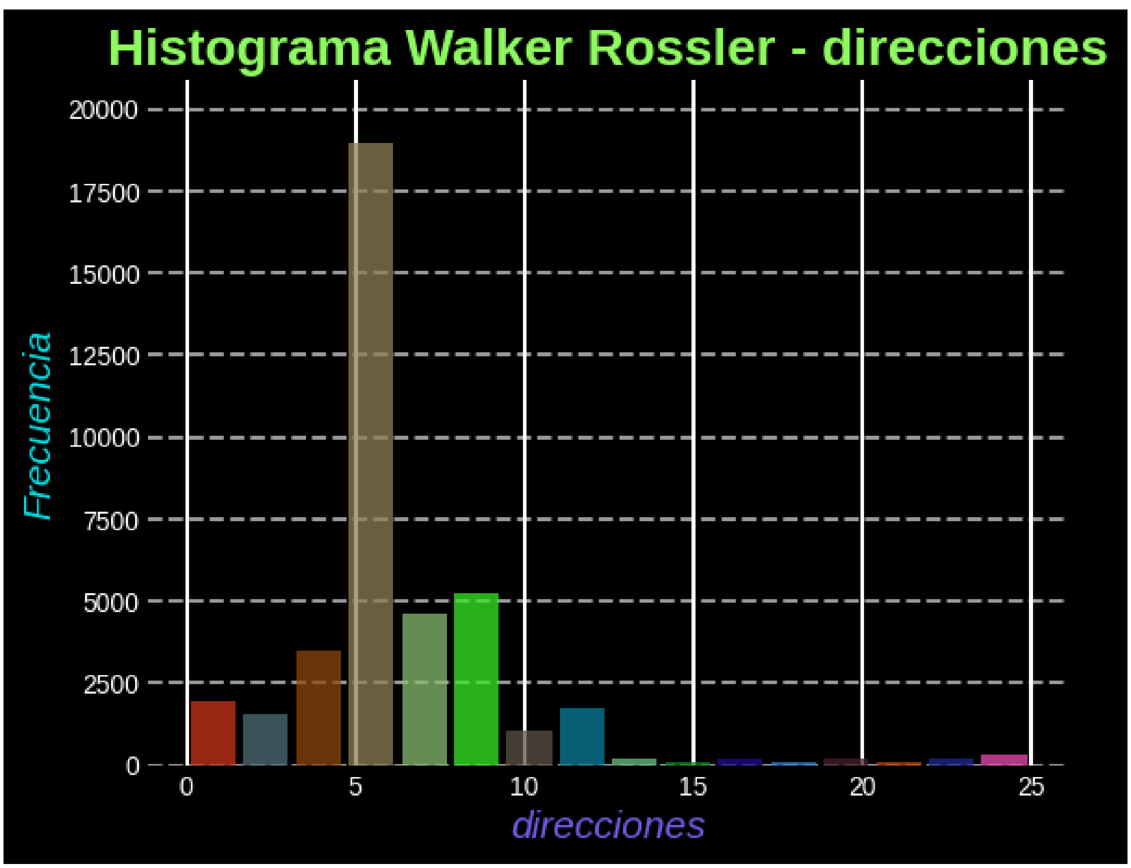
1.8 Atractor Rossler



1.8.1 Direcciones

[44]:
```
modelo = "transfrossler"
metrica = "direcciones"
folder = "Rossler"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
 ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
 ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```


Caminate de Rossler - direcciones
Valor de x
Tasa de crecimiento t
0
5
10
15
20
25
0
5000
10000
15000
20000
25000
30000
35000
40000


Histograma Walker Rossler - direcciones
Frecuencia
direcciones
0
2500
5000
7500
10000
12500
15000
17500
20000
0
5
10
15
20
25

Interpretación de las Métricas Media (6.525): La media indica que, en promedio, las direcciones seleccionadas por el caminante están alrededor del índice 6.525.

Mediana (6.0): La mediana sugiere que la mitad de las direcciones seleccionadas son menores o iguales a 6 y la otra mitad son mayores o iguales a 6.

Moda (6, 13919): La moda muestra que la dirección 6 es la más frecuente, seleccionada 13,919 veces, lo que indica una fuerte preferencia del caminante por moverse en esta dirección.

Desviación Estándar (3.421): La desviación estándar de 3.421 indica que las direcciones seleccionadas varían considerablemente alrededor de la media.

Varianza (11.704): La varianza de 11.704 confirma una alta dispersión en las direcciones seleccionadas.

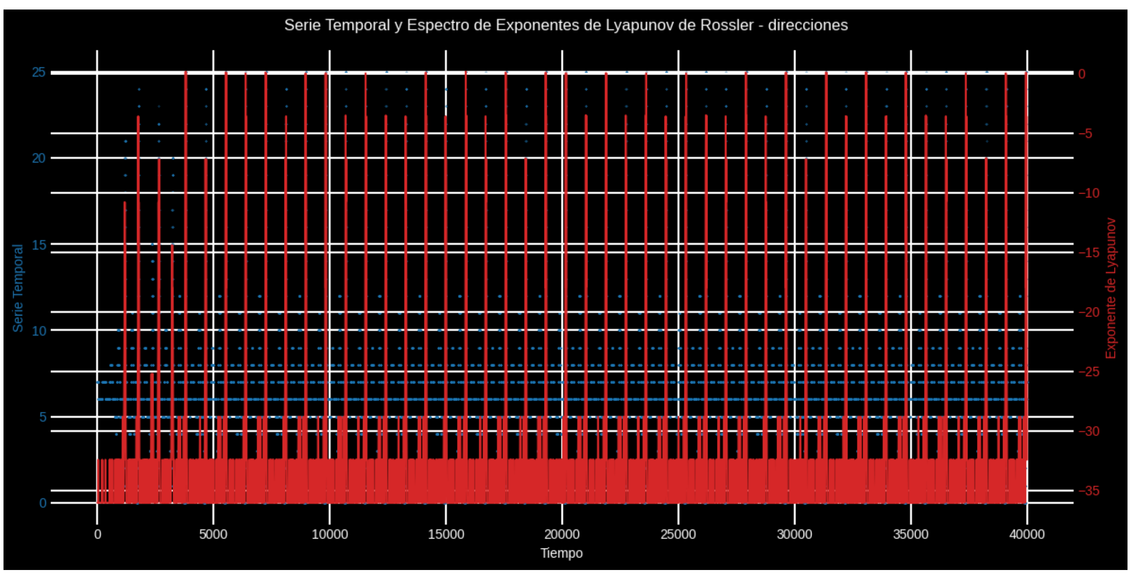
Asimetría (2.231): La asimetría de 2.231 sugiere que la distribución está sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (9.151): La curtosis de 9.151 indica que la distribución es más puntiaguda en comparación con una distribución normal, sugiriendo más valores extremos.

Entropía (2.237): La entropía de 2.237 indica un nivel moderado de incertidumbre o aleatoriedad en las direcciones seleccionadas por el caminante.

1.8.2 Exponentes de lyapounov y Kaplan Yorke

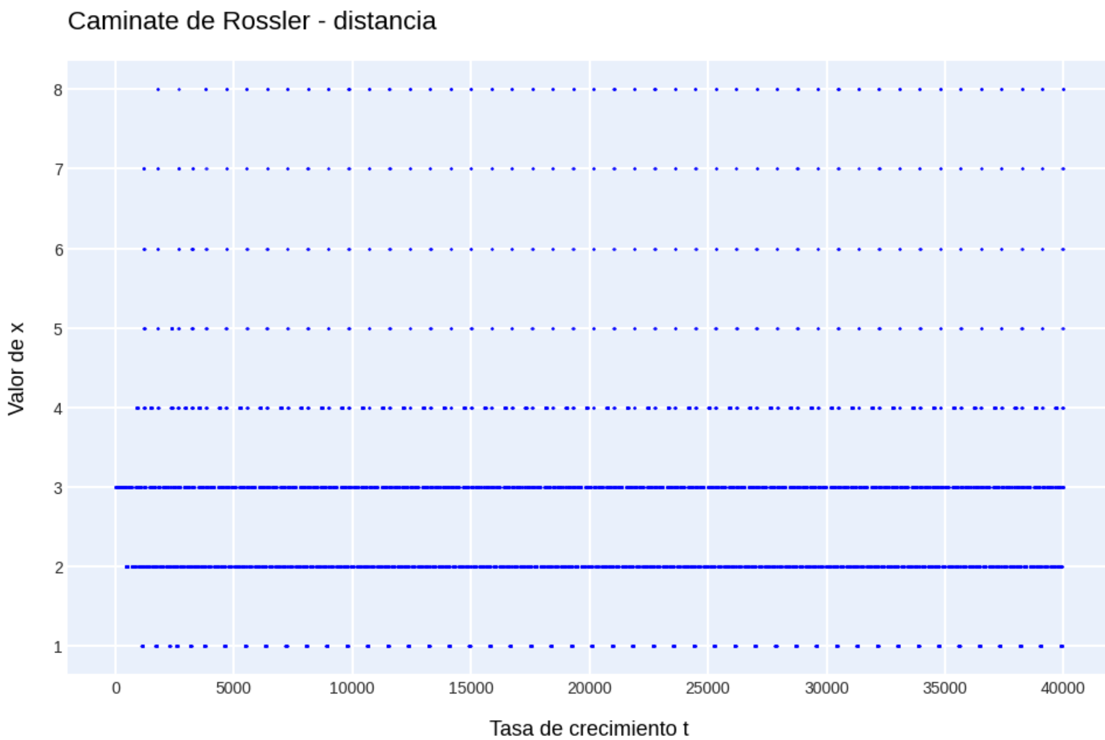
[45]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
  ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```

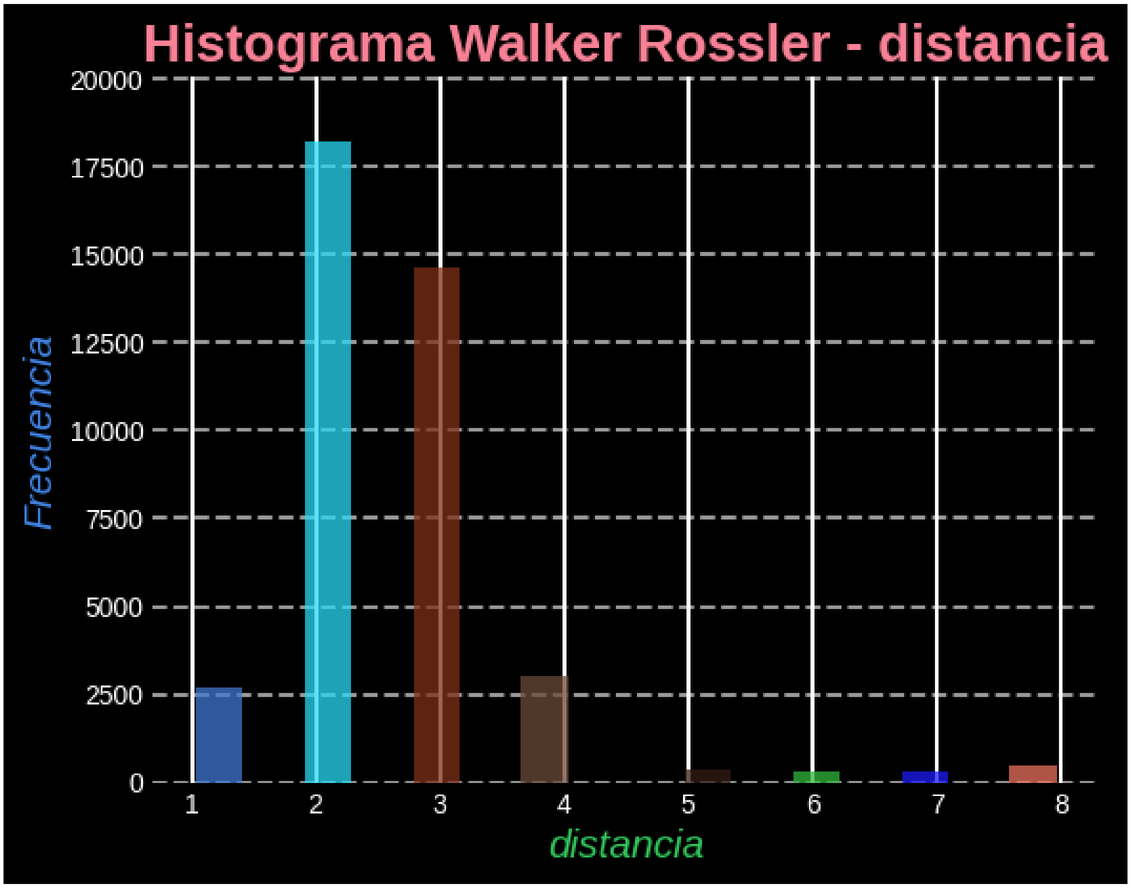


```
La dimension de Kaplan York es: 6.0
```

1.8.3 Distancias

[46]:
```
metrica = "distancia"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
 ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
 ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```





Interpretación de las Métricas Media (2.615): La media indica que, en promedio, el caminante avanza aproximadamente 2.615 unidades por movimiento.

Mediana (2.0): La mediana sugiere que la mitad de las distancias recorridas son menores o iguales a 2 y la otra mitad son mayores o iguales a 2.

Moda (2, 18249): La moda muestra que la distancia de 2 unidades es la más frecuente, seleccionada 18,249 veces, lo que indica una fuerte preferencia por esta distancia.

Desviación Estándar (1.081): La desviación estándar de 1.081 sugiere una baja variabilidad en las distancias recorridas.

Varianza (1.168): La varianza de 1.168 confirma la observación de la desviación estándar, indicando una baja dispersión en las distancias recorridas.

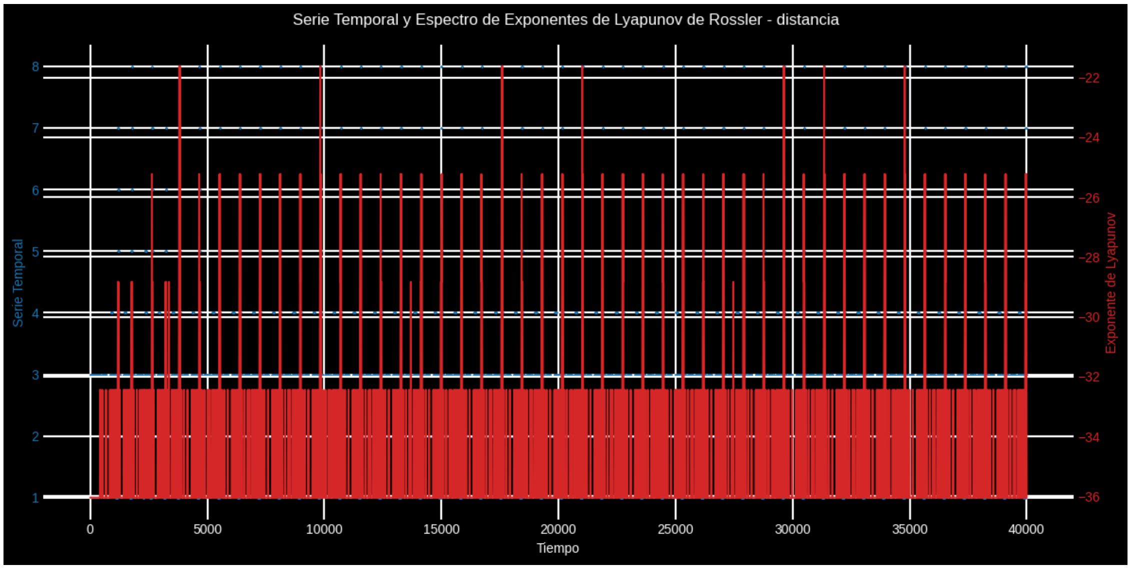
Asimetría (2.152): La asimetría de 2.152 sugiere que la distribución está sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (7.807): La curtosis de 7.807 indica que la distribución es más puntiaguda en comparación con una distribución normal, sugiriendo más valores extremos.

Entropía (1.27): La entropía de 1.27 indica un nivel bajo de incertidumbre en la selección de distancias.

1.8.4 Exponentes de lyapounov y Kaplan Yorke

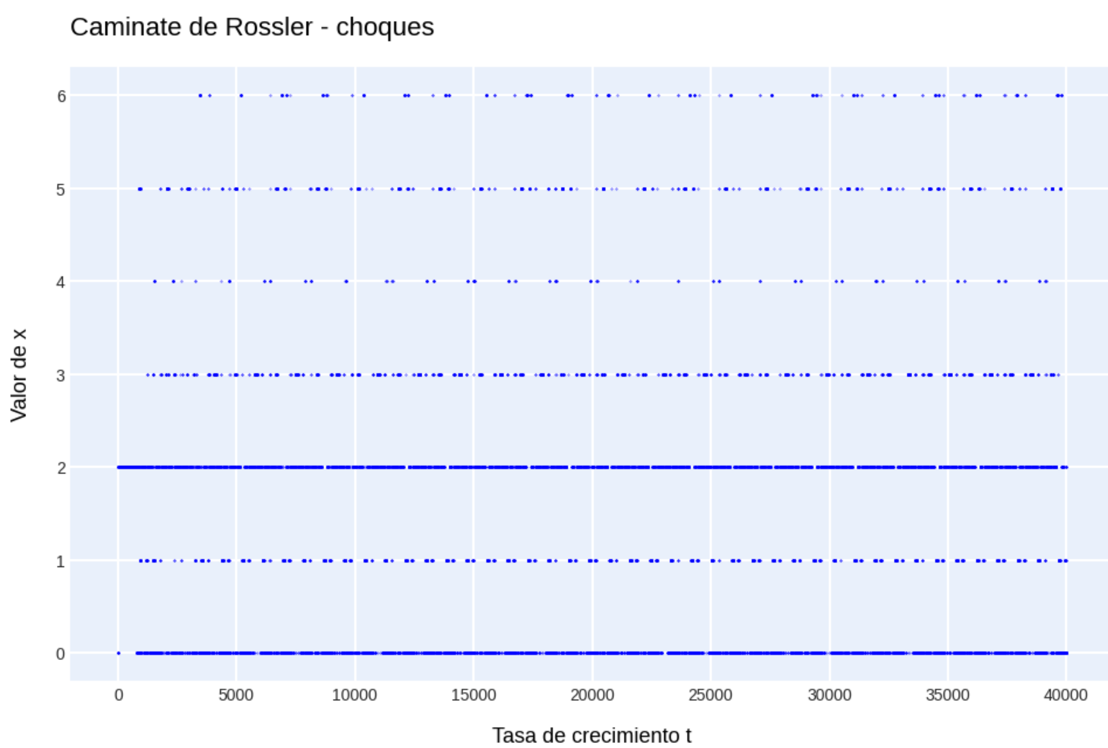
[47]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```



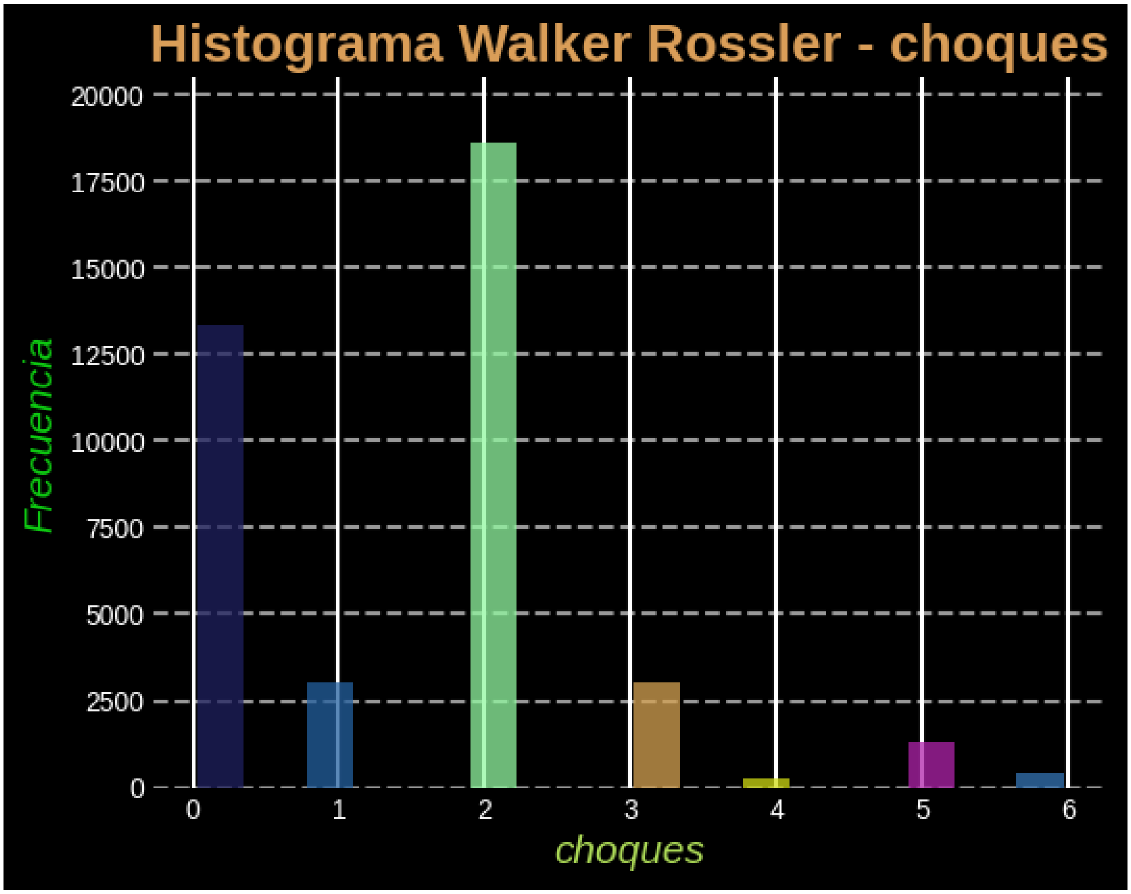

```
La dimension de Kaplan York es: 0.0
```

1.8.5 Choques

[48]:
```
metrica = "choques"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```



Caminate de Rossler - choques
Valor de x
0
1
2
3
4
5
6
0
5000
10000
15000
20000
25000
30000
35000
40000
Tasa de crecimiento t



Interpretación de las Métricas Media (1.485): La media indica que, en promedio, el caminante choca con las paredes del cubo aproximadamente 1.485 veces por movimiento.

Mediana (2.0): La mediana sugiere que la mitad de los movimientos resultan en choques con la pared 2 o menor.

Moda (2, 18638): La moda muestra que la pared 2 es la más frecuente, ocurriendo 18,638 veces.

Desviación Estándar (1.307): La desviación estándar de 1.307 sugiere una variabilidad considerable en la pared de choque.

Varianza (1.709): La varianza de 1.709 confirma una alta dispersión en la pared de choque.

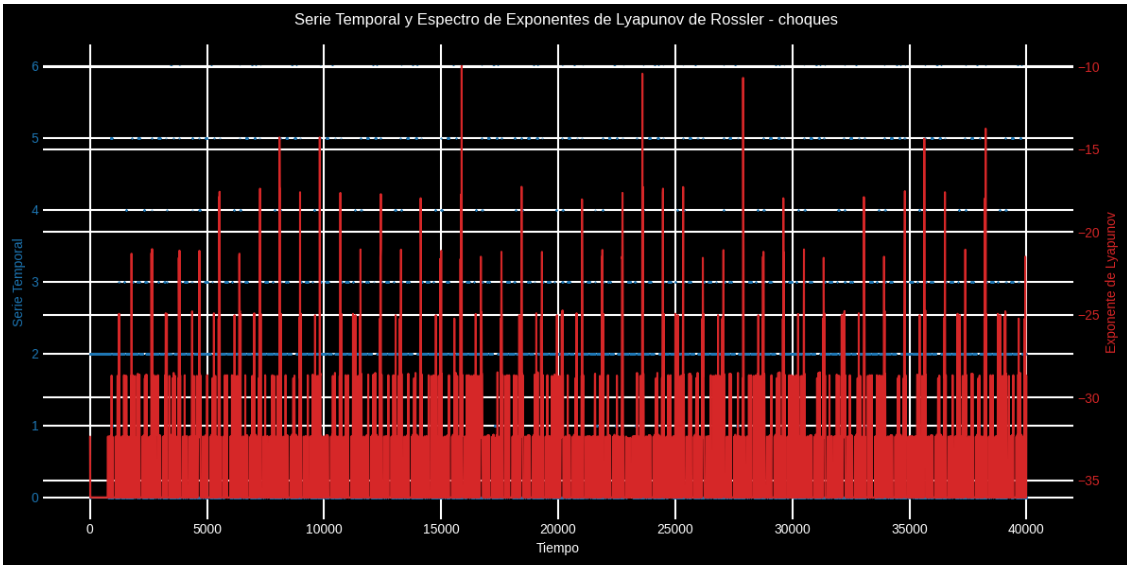
Asimetría (0.765): La asimetría de 0.765 sugiere que la distribución está sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (0.987): La curtosis de 0.987 indica que la distribución es ligeramente más puntiaguda en comparación con una distribución normal, sugiriendo más valores extremos.

Entropía (1.303): La entropía de 1.303 indica un nivel moderado de incertidumbre en la ocurrencia de choques.

1.8.6 Exponentes de lyapounov y Kaplan yORKE

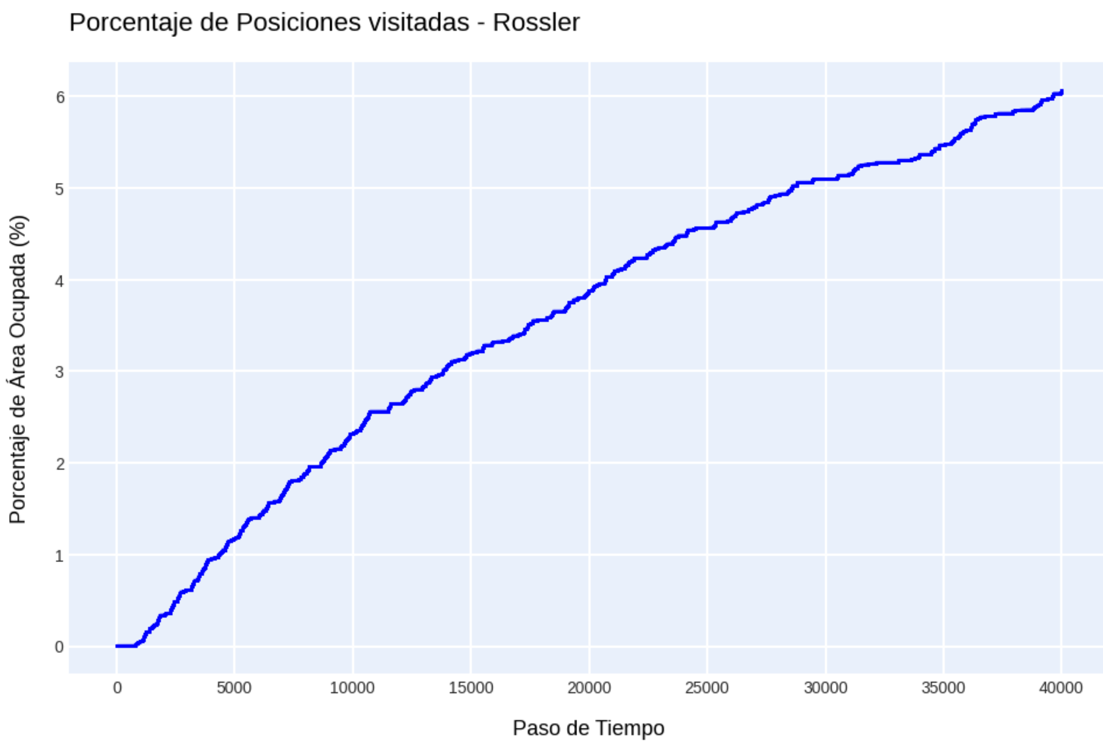
[49]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
    ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```




```
La dimension de Kaplan York es: 0.09080328696532036
```

1.8.7 Posiciones

[50]:
```
metrica = "posiciones"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ";")

graficar_posicionesOcupadas(49, walker_feig, title=f"Porcentaje de Posiciones␣
    ↪visitadas - {folder}")
ex = DistribucionProbabilidadVector(np.array(walker_feig))
res = ex.calcular_metricas()
```



Interpretación de las Métricas Media (39.776, 27.206, 14.845): La media indica que, en promedio, las posiciones visitadas por el caminante están alrededor de estos valores en el espacio tridimensional, con un sesgo hacia ciertas áreas del cubo.

Mediana (49.0, 28.0, 13.0): La mediana sugiere que la mitad de las posiciones visitadas se encuentran por debajo de estos valores, indicando una concentración en ciertas áreas del cubo.

Moda ([49, 49, 0], [23332, 3995, 11614]): La moda muestra que ciertas posiciones específicas son visitadas con mucha más frecuencia que otras, lo que indica una fuerte preferencia o patrón repetitivo en el movimiento del caminante.

Desviación Estándar (15.166, 14.486, 14.605): La desviación estándar indica una considerable variabilidad en las posiciones visitadas, especialmente en la primera y segunda dimensiones.

Varianza (230.014, 209.857, 213.317): La varianza confirma la alta dispersión en las posiciones visitadas, con una mayor dispersión en la primera dimensión.

Asimetría (-1.509, -0.262, 0.758): La asimetría sugiere que la distribución de las posiciones visitadas tiene diferentes sesgos en cada dimensión, con una distribución más equilibrada en la segunda dimensión y distribuciones sesgadas hacia valores menores en las otras dos dimensiones.

Coeficiente de Variación (0.381, 0.532, 0.984): El coeficiente de variación indica que hay una relativa consistencia en la exploración a lo largo de las dimensiones, con la primera y segunda dimensiones mostrando más variabilidad relativa.

Curtosis (0.838, -0.733, -0.538): La curtosis negativa en las últimas dos dimensiones indica que las posiciones visitadas tienen distribuciones más planas en comparación con una distribución normal, sugiriendo menos valores extremos.

Entropía (2.205, 3.523, 3.138): La entropía indica un nivel moderado a alto de incertidumbre en la exploración de posiciones, lo que sugiere un balance entre aleatoriedad y repetición en el movimiento del caminante.

1.8.8 Serie Temporal y Espectro de Exponentes de Lyapunov - Rossler (direcciones)

- **Serie Temporal**:
  - **Descripción**: La gráfica muestra la evolución de las direcciones seleccionadas por el caminante aleatorio a lo largo del tiempo.
  - **Observaciones**: Las direcciones se concentran en un rango de aproximadamente 0 a 25, con patrones repetitivos y picos altos.
  - **Interpretación**: Esto sugiere un comportamiento cíclico con direcciones que varían de manera predecible en ciertos intervalos.
- **Exponentes de Lyapunov**:
  - **Descripción**: Los exponentes de Lyapunov están superpuestos a la serie temporal de direcciones.
  - **Observaciones**: Predominan los exponentes negativos.
  - **Interpretación**: La presencia predominante de exponentes negativos indica estabilidad local en muchas regiones, con un comportamiento menos caótico.
- **Dimensión de Kaplan-Yorke**:
  - **Valor**: 6.0
  - **Interpretación**: Este valor indica una complejidad alta en la dinámica del sistema, con una estructura fractal más desarrollada.

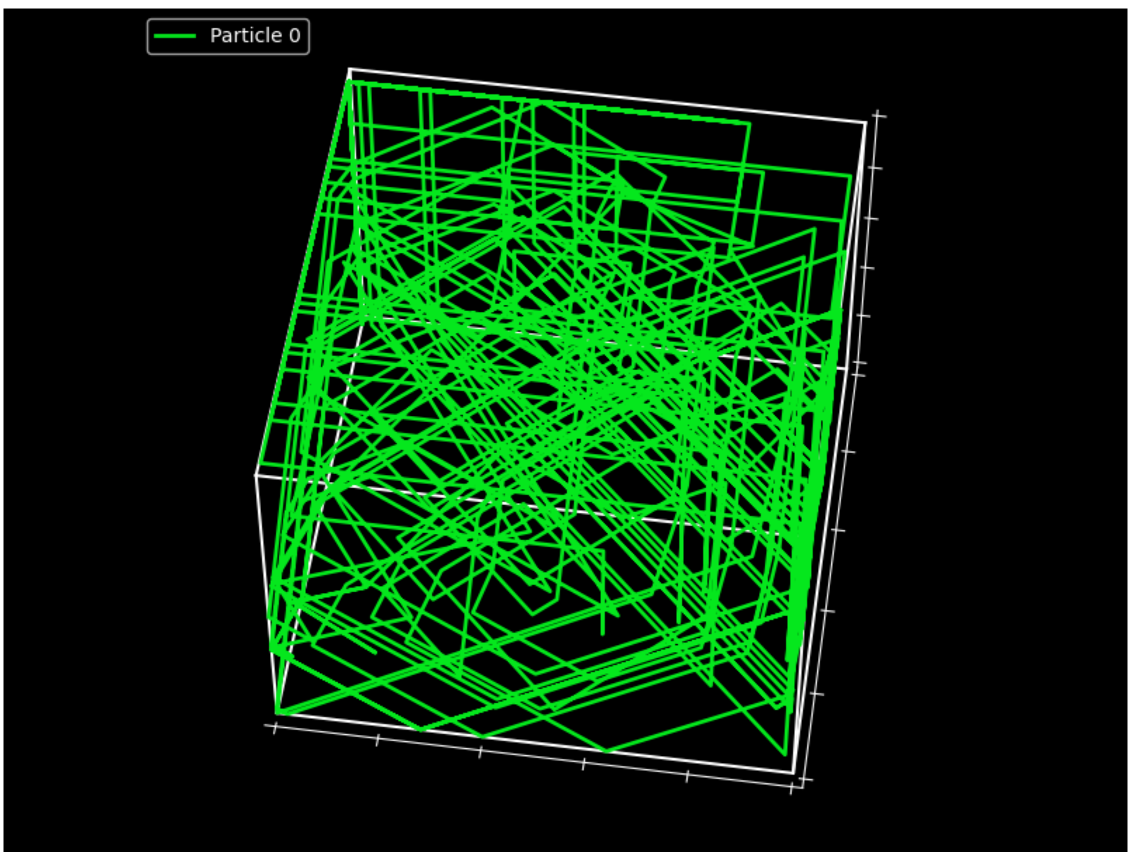
1.8.9 Serie Temporal y Espectro de Exponentes de Lyapunov - Rossler (distancia)

- **Serie Temporal**:
  - **Descripción**: La gráfica muestra la evolución de las distancias recorridas en cada paso por el caminante aleatorio a lo largo del tiempo.
  - **Observaciones**: Las distancias se concentran principalmente en valores entre 1 y 8, con picos regulares.
  - **Interpretación**: Esto sugiere que el caminante realiza desplazamientos pequeños y medianos de manera predecible.
- **Exponentes de Lyapunov**:
  - **Descripción**: Los exponentes de Lyapunov están superpuestos a la serie temporal de distancias.
  - **Observaciones**: Predominan los exponentes negativos.
  - **Interpretación**: La presencia de exponentes negativos refuerza la idea de estabilidad local, con un comportamiento menos caótico.
- **Dimensión de Kaplan-Yorke**:
  - **Valor**: 0.0
  - **Interpretación**: Este valor indica una falta de complejidad en la dinámica del sistema, sugiriendo un comportamiento más regular y predecible.

1.8.10 Serie Temporal y Espectro de Exponentes de Lyapunov - Rossler (choques)

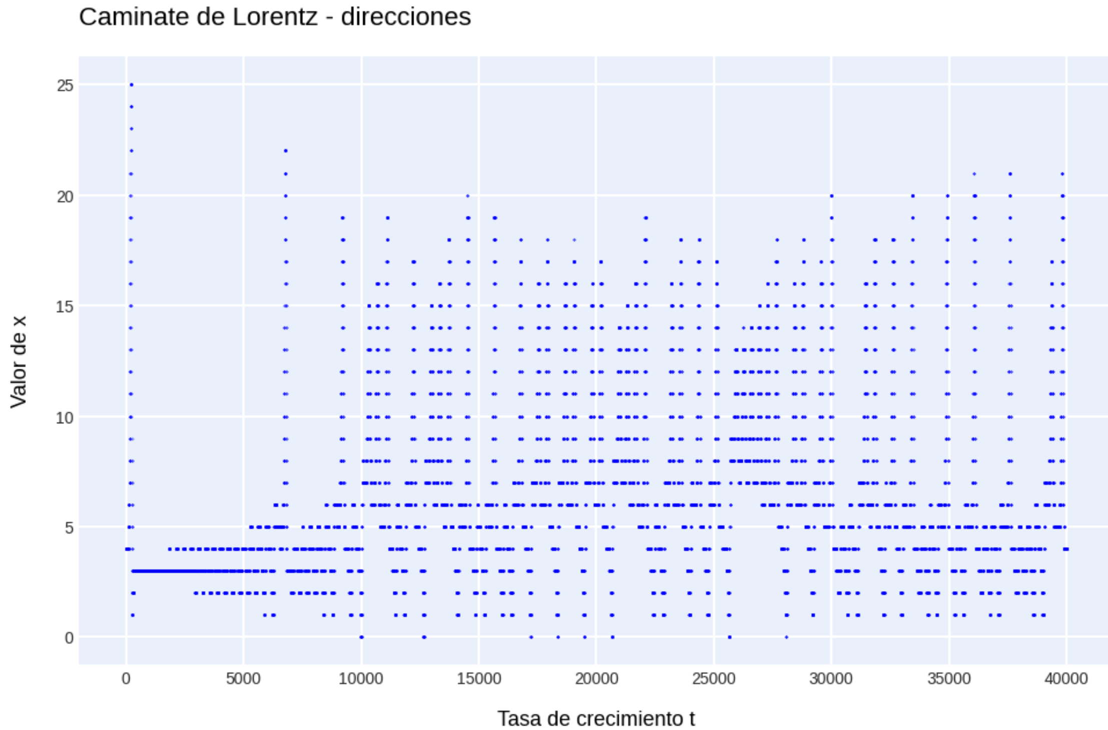
- **Serie Temporal**:
  - **Descripción**: La gráfica muestra la evolución de los choques (colisiones) del caminante aleatorio a lo largo del tiempo.
  - **Observaciones**: Los choques se concentran en un rango de aproximadamente 0 a 6, con alta variabilidad.
  - **Interpretación**: Esto indica un comportamiento impredecible y variado en términos de choques.
- **Exponentes de Lyapunov**:
  - **Descripción**: Los exponentes de Lyapunov están superpuestos a la serie temporal de choques.
  - **Observaciones**: Predominan los exponentes negativos.
  - **Interpretación**: La presencia predominante de exponentes negativos indica estabilidad local en muchas regiones, aunque con comportamiento caótico.
- **Dimensión de Kaplan-Yorke**:
  - **Valor**: 0.09080328696532036
  - **Interpretación**: Este valor sugiere una menor complejidad en la dinámica del sistema comparado con las gráficas de direcciones, aunque todavía presenta cierta estructura fractal y comportamiento caótico.

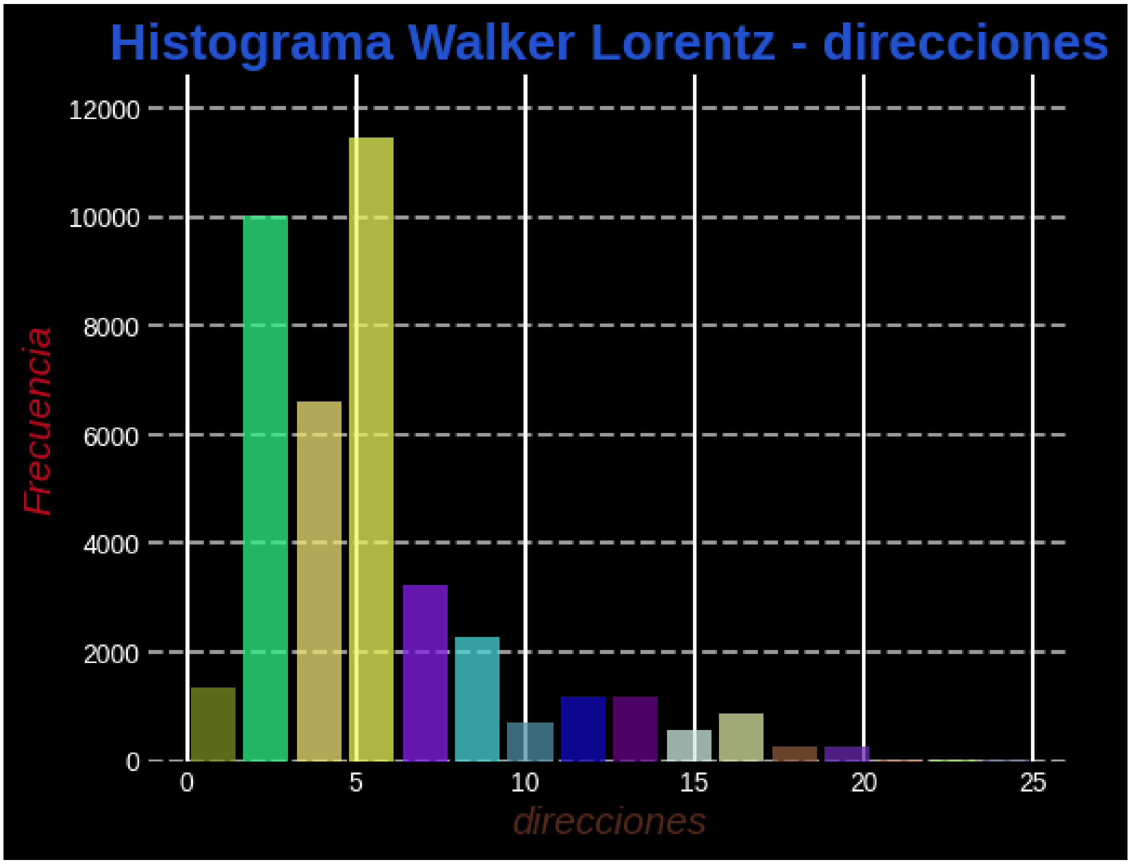
1.9 Atractor Lorentz



1.9.1 Direcciones

[51]:
```
modelo = "transflorentz"
metrica = "direcciones"
folder = "Lorentz"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
    ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
    ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```





Interpretación de las Métricas Media (5.737): La media indica que, en promedio, las direcciones seleccionadas por el caminante están alrededor del índice 5.737.

Mediana (5.0): La mediana sugiere que la mitad de las direcciones seleccionadas son menores o iguales a 5 y la otra mitad son mayores o iguales a 5.

Moda (3, 7023): La moda muestra que la dirección 3 es la más frecuente, seleccionada 7,023 veces, lo que indica una fuerte preferencia del caminante por moverse en esta dirección.

Desviación Estándar (3.718): La desviación estándar de 3.718 indica que las direcciones seleccionadas varían considerablemente alrededor de la media.

Varianza (13.823): La varianza de 13.823 confirma una alta dispersión en las direcciones seleccionadas.

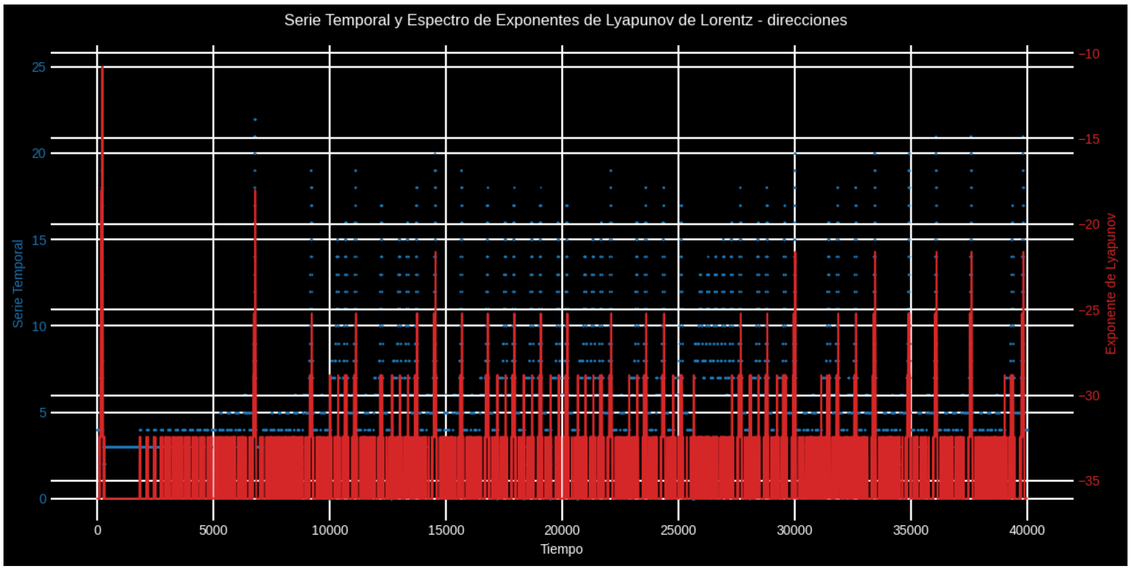
Asimetría (1.682): La asimetría de 1.682 sugiere que la distribución está sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (2.897): La curtosis de 2.897 indica que la distribución es más puntiaguda en comparación con una distribución normal, sugiriendo más valores extremos.

Entropía (2.44): La entropía de 2.44 indica un nivel moderado de incertidumbre o aleatoriedad en las direcciones seleccionadas por el caminante.

1.9.2 Exponentes de lyapounov Y kAPLAN yORKE

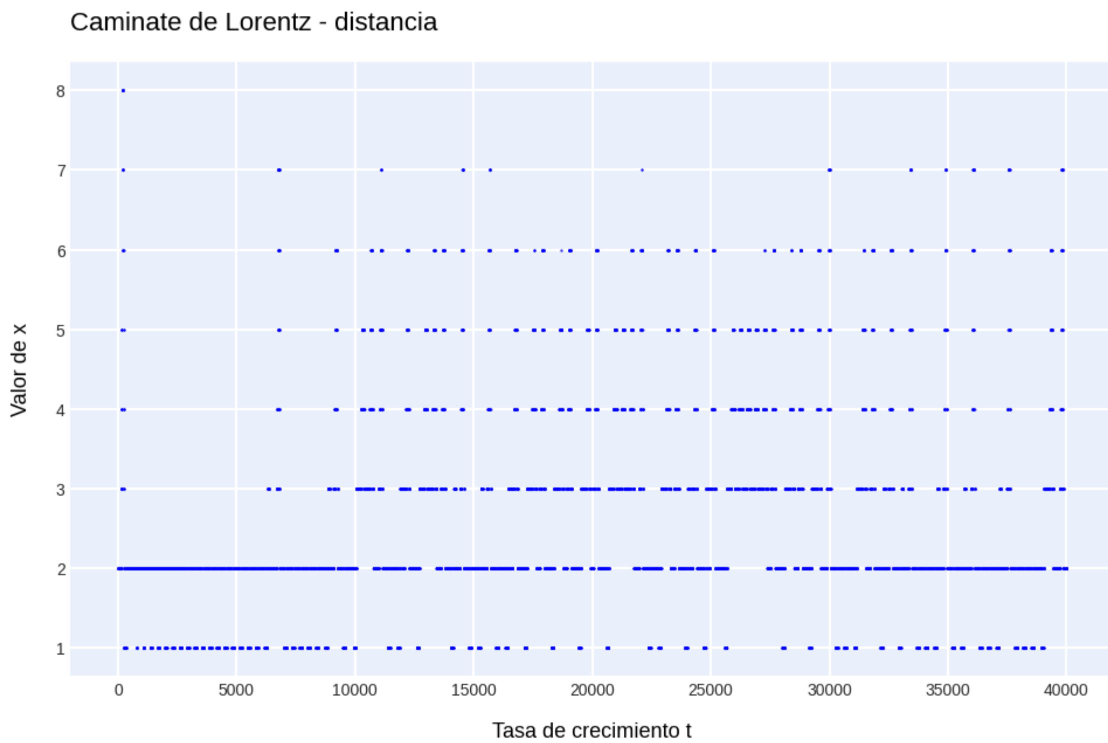
[52]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
    ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```



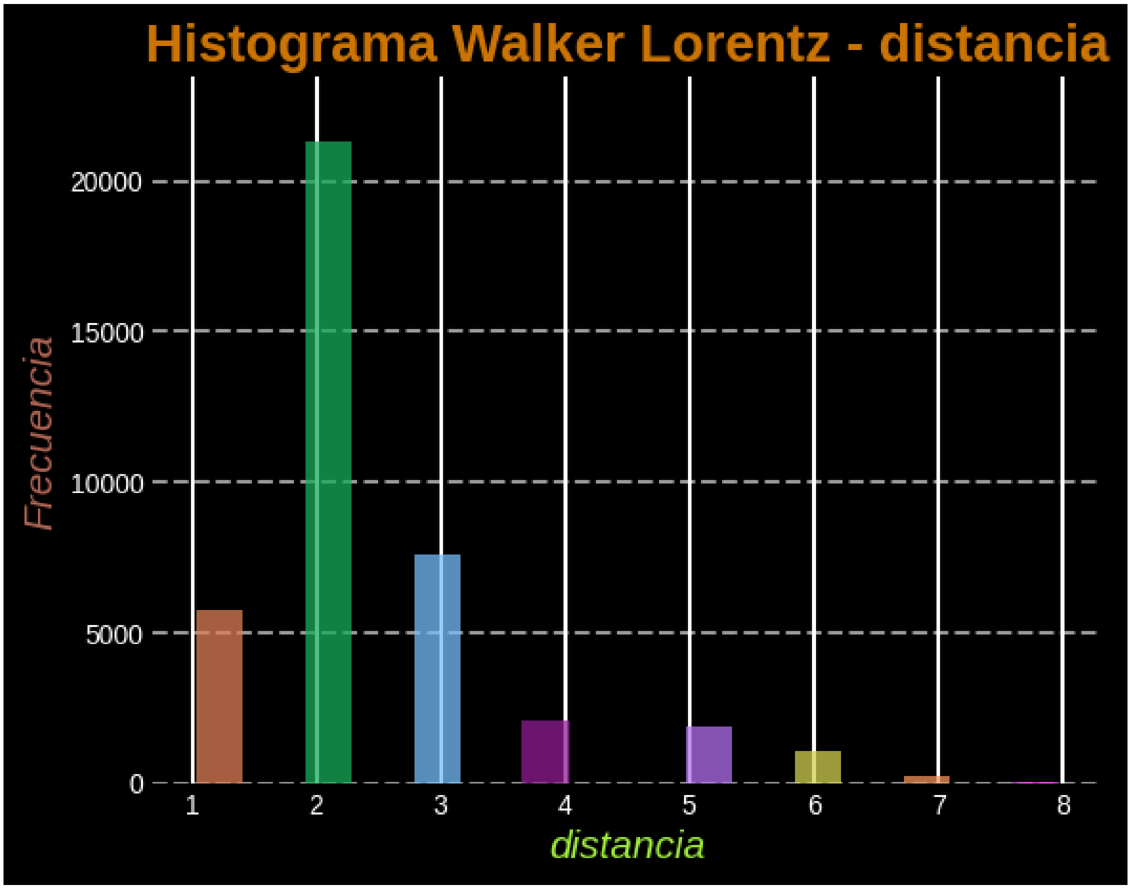

```
La dimension de Kaplan York es: 0.5999999999999999
```

1.9.3 Distancias

[53]:
```
metrica = "distancia"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
    ↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
    ↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```



Caminate de Lorentz - distancia
Valor de x
8
7
6
5
4
3
2
1
0
5000
10000
15000
20000
25000
30000
35000
40000
Tasa de crecimiento t



Interpretación de las Métricas Media (2.431): La media indica que, en promedio, el caminante avanza aproximadamente 2.431 unidades por movimiento.

Mediana (2.0): La mediana sugiere que la mitad de las distancias recorridas son menores o iguales a 2 y la otra mitad son mayores o iguales a 2.

Moda (2, 21332): La moda muestra que la distancia de 2 unidades es la más frecuente, seleccionada 21,332 veces, lo que indica una fuerte preferencia por esta distancia.

Desviación Estándar (1.175): La desviación estándar de 1.175 sugiere una baja variabilidad en las distancias recorridas.

Varianza (1.38): La varianza de 1.38 confirma la observación de la desviación estándar, indicando una baja dispersión en las distancias recorridas.

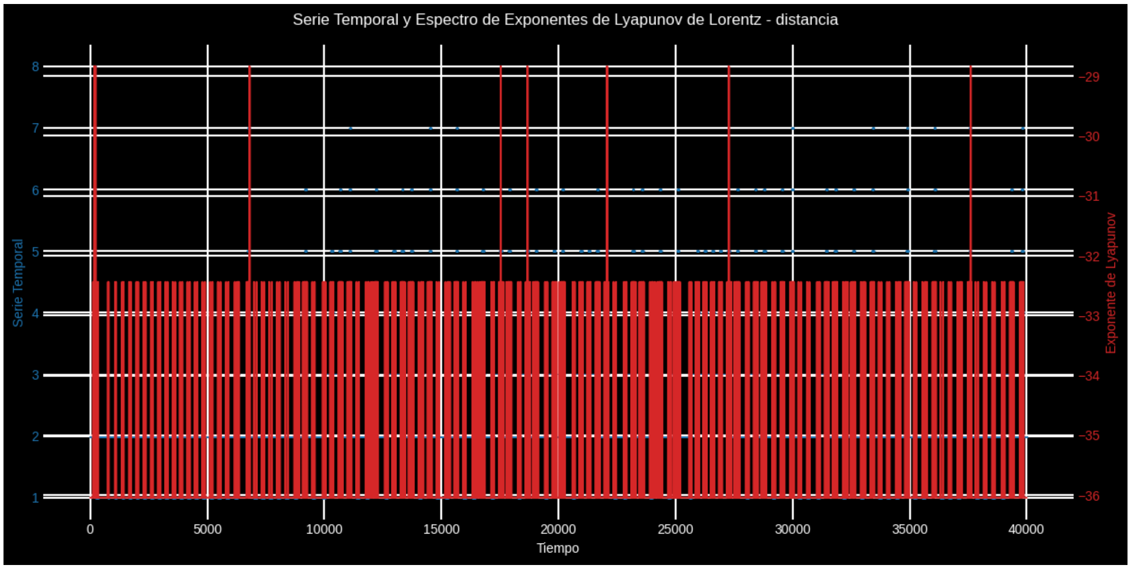
Asimetría (1.52): La asimetría de 1.52 sugiere que la distribución está sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (2.547): La curtosis de 2.547 indica que la distribución es más puntiaguda en comparación con una distribución normal, sugiriendo más valores extremos.

Entropía (1.359): La entropía de 1.359 indica un nivel bajo de incertidumbre en la selección de distancias.

1.9.4 Exponentes de lyapounov y Kaplan Yorke

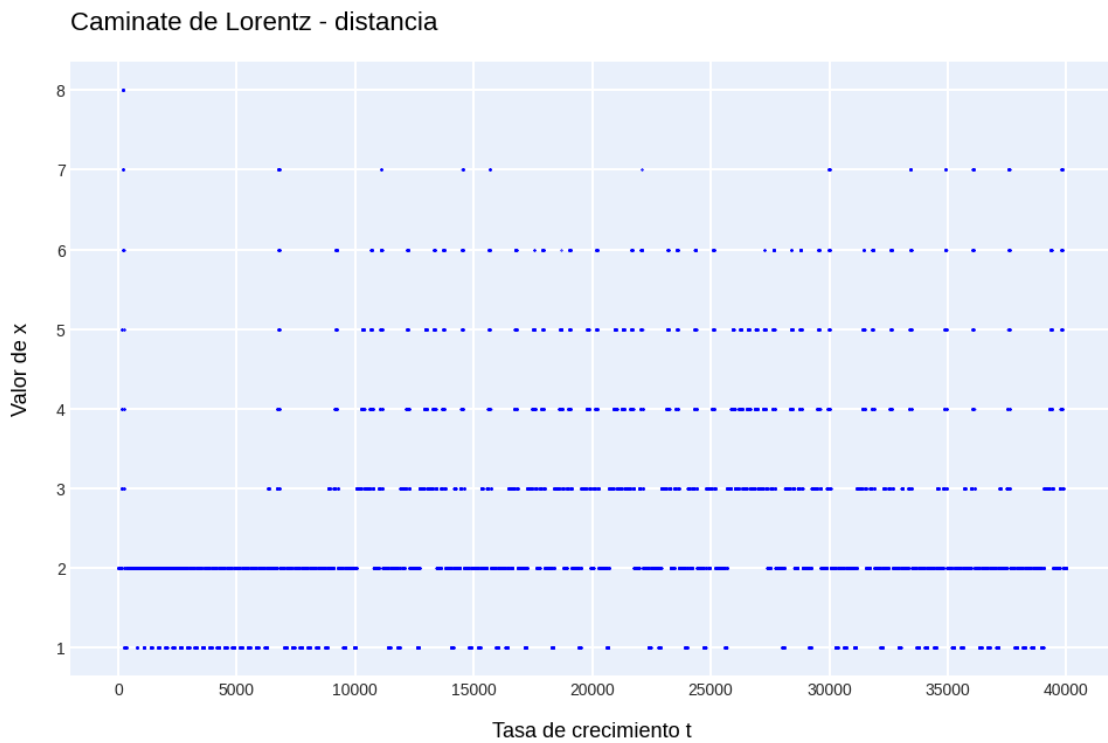
[54]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```



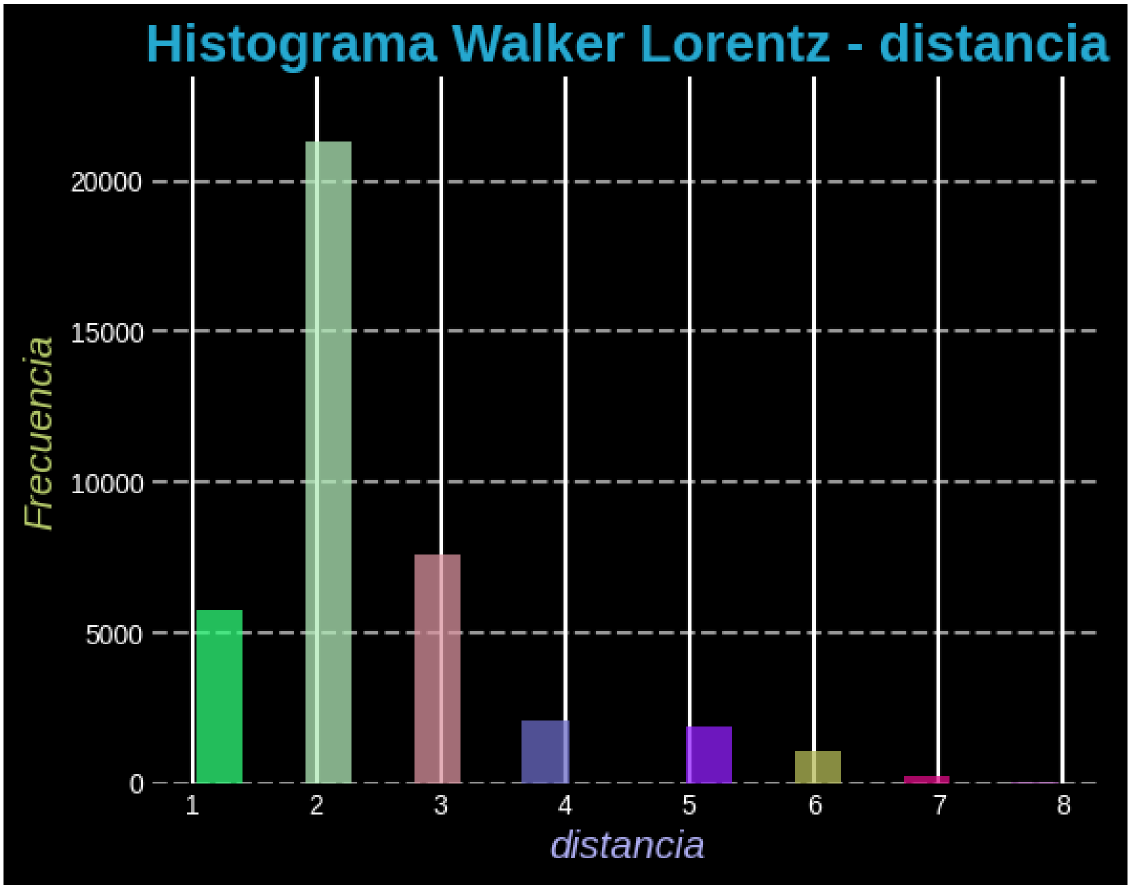

```
La dimension de Kaplan York es: 0.0
```

1.9.5 Choques

[55]:
```
### Exponentes de lyapounovmetrica = "choques"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ",")
tiempo = range(1, 40001)
graficar(walker_feig, tiempo, width=7, height=5 ,titulo=f"Caminate de {folder}␣
↪- {metrica}")
plotear_hist(walker_feig, f"Histograma Walker {folder} - {metrica}", metrica,␣
↪"Frecuencia")
ex = DistribucionProbabilidad(walker_feig)
res = ex.calcular_metricas([25, 50, 75], [1, 2, 3])
```



Caminate de Lorentz - distancia
Valor de x
Tasa de crecimiento t
8
7
6
5
4
3
2
1
0
5000
10000
15000
20000
25000
30000
35000
40000



Interpretación de las Métricas Media (2.077): La media indica que, en promedio, el caminante choca con las paredes del cubo aproximadamente 2.077 veces por movimiento.

Mediana (2.0): La mediana sugiere que la mitad de los movimientos resultan en choques con la pared 2 o menor.

Moda (2, 12580): La moda muestra que la pared 2 es la más frecuente, ocurriendo 12,580 veces.

Desviación Estándar (1.495): La desviación estándar de 1.495 sugiere una variabilidad considerable en la pared de choque.

Varianza (2.234): La varianza de 2.234 confirma una alta dispersión en la pared de choque.

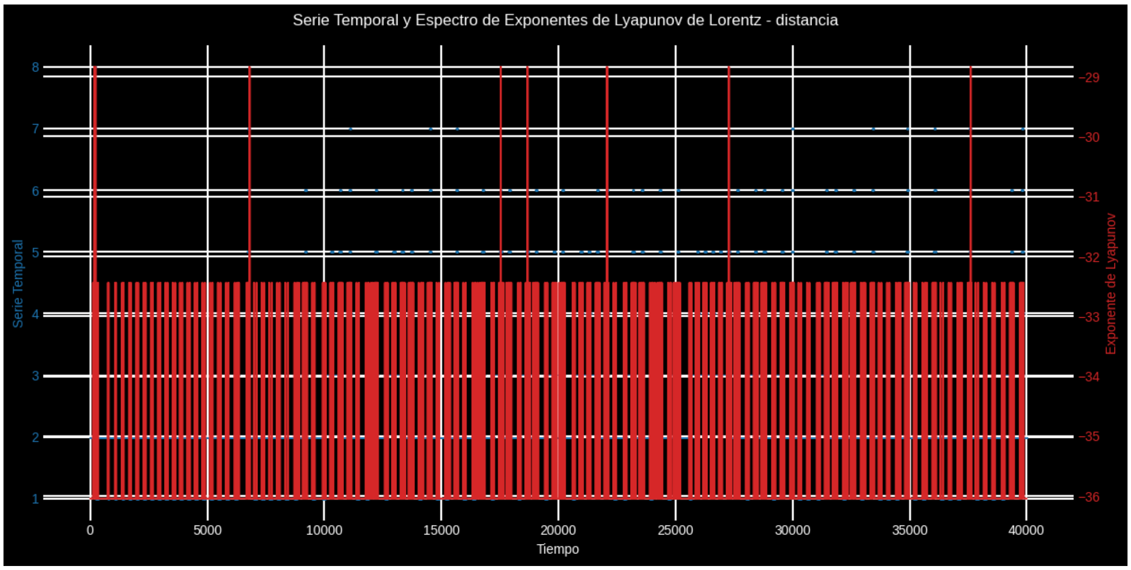
Asimetría (0.283): La asimetría de 0.283 sugiere que la distribución está ligeramente sesgada hacia la derecha, indicando una cola más larga en el lado derecho de la distribución.

Curtosis (0.079): La curtosis de 0.079 indica que la distribución es casi normal, sugiriendo valores extremos similares a una distribución normal.

Entropía (1.502): La entropía de 1.502 indica un nivel moderado de incertidumbre en la ocurrencia de choques.

1.9.6 Exponentes de lyapounov y Kaplan yORKE

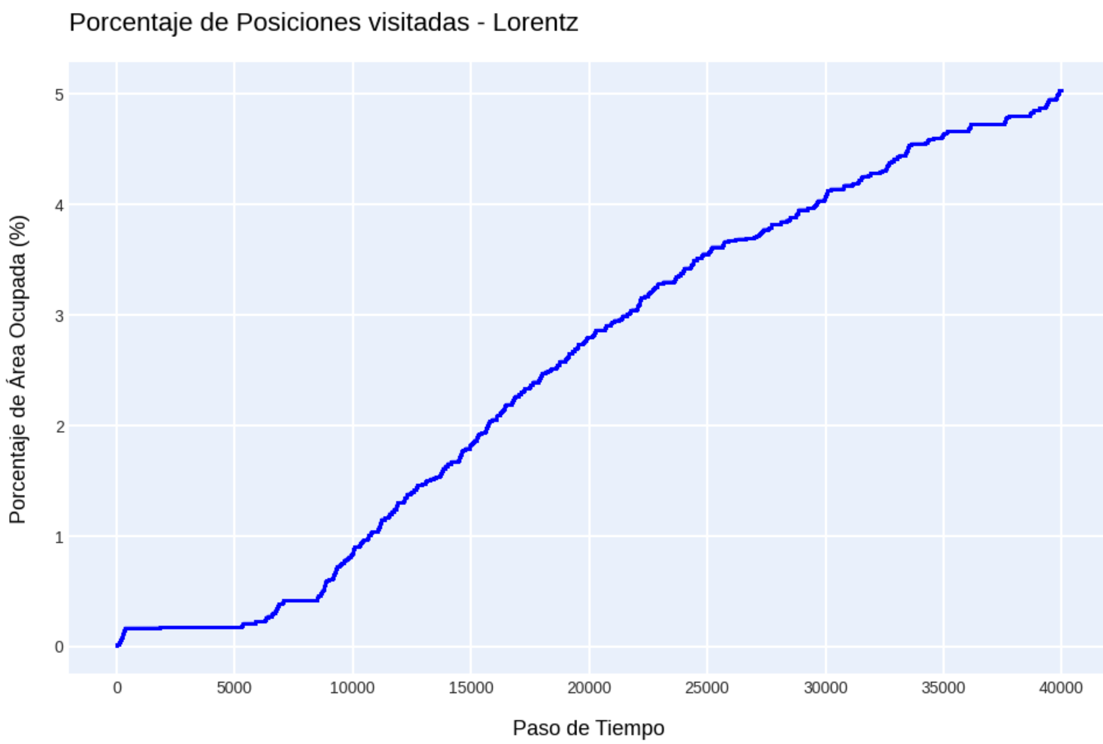
[56]:
```
window_size = 10
lyapunov_exponents = plot_lyapunov_spectrum(walker_feig, window_size, folder,␣
 ↪metrica)
dim_kaplan_yorke = dimension_kaplan_yorke(lyapunov_exponents)
print(f"La dimension de Kaplan York es: {dim_kaplan_yorke}")
```




```
La dimension de Kaplan York es: 0.0
```

1.9.7 Posiciones

[57]:
```
metrica = "posiciones"
walker_feig = cargar_csv(f"datos/{folder}/{metrica}_{modelo}0.txt", ";")

graficar_posicionesOcupadas(49, walker_feig, title=f"Porcentaje de Posiciones␣
 ↪visitadas - {folder}")
ex = DistribucionProbabilidadVector(np.array(walker_feig))
res = ex.calcular_metricas()
```



Interpretación de las Métricas Media (24.814, 32.707, 29.064): La media indica que, en promedio, las posiciones visitadas por el caminante están alrededor de estos valores en el espacio tridimensional, con un sesgo hacia ciertas áreas del cubo.

Mediana (27.0, 35.0, 32.0): La mediana sugiere que la mitad de las posiciones visitadas se encuentran por debajo de estos valores, indicando una concentración en ciertas áreas del cubo.

Moda ([0, 49, 49], [14650, 15071, 14979]): La moda muestra que ciertas posiciones específicas son visitadas con mucha más frecuencia que otras, lo que indica una fuerte preferencia o patrón repetitivo en el movimiento del caminante.

Desviación Estándar (21.873, 15.764, 19.215): La desviación estándar indica una considerable variabilidad en las posiciones visitadas, especialmente en la primera y tercera dimensiones.

Varianza (478.427, 248.493, 369.233): La varianza confirma la alta dispersión en las posiciones visitadas, con una mayor dispersión en la primera dimensión.

Asimetría (-0.046, -0.413, -0.34): La asimetría sugiere que la distribución de las posiciones visitadas tiene diferentes sesgos en cada dimensión, con una distribución más equilibrada en la segunda dimensión y distribuciones ligeramente sesgadas hacia valores menores en las otras dos dimensiones.

Coeficiente de Variación (0.881, 0.482, 0.661): El coeficiente de variación indica que hay una relativa consistencia en la exploración a lo largo de las dimensiones, con la primera y tercera dimensiones mostrando más variabilidad relativa.

Curtosis (-1.801, -1.146, -1.44): La curtosis negativa en todas las dimensiones indica que las posiciones visitadas tienen distribuciones más planas en comparación con una distribución normal, sugiriendo menos valores extremos.

Entropía (2.148, 2.894, 2.529): La entropía indica un nivel moderado de incertidumbre en la exploración de posiciones, lo que sugiere un balance entre aleatoriedad y repetición en el movimiento del caminante.

1.9.8 Serie Temporal y Espectro de Exponentes de Lyapunov - Lorentz (direcciones)

- **Serie Temporal**:
  - **Descripción**: La gráfica muestra la evolución de las direcciones seleccionadas por el caminante aleatorio a lo largo del tiempo.
  - **Observaciones**: Las direcciones se concentran en un rango de aproximadamente 0 a 25, con patrones de picos altos y variabilidad.
  - **Interpretación**: Esto sugiere un comportamiento cíclico y variado en términos de direcciones tomadas por el caminante.
- **Exponentes de Lyapunov**:
  - **Descripción**: Los exponentes de Lyapunov están superpuestos a la serie temporal de direcciones.
  - **Observaciones**: Predominan los exponentes negativos.
  - **Interpretación**: La presencia predominante de exponentes negativos indica estabilidad local en muchas regiones, aunque con un comportamiento caótico.
- **Dimensión de Kaplan-Yorke**:
  - **Valor**: 0.5999999999999999
  - **Interpretación**: Este valor indica una baja complejidad en la dinámica del sistema, con una estructura fractal menos desarrollada.

1.9.9 Serie Temporal y Espectro de Exponentes de Lyapunov - Lorentz (distancia)

- **Serie Temporal**:
  - **Descripción**: La gráfica muestra la evolución de las distancias recorridas en cada paso por el caminante aleatorio a lo largo del tiempo.
  - **Observaciones**: Las distancias se concentran principalmente en valores entre 1 y 8, con patrones repetitivos y predecibles.
  - **Interpretación**: Esto sugiere que el caminante realiza desplazamientos pequeños y medianos de manera regular.
- **Exponentes de Lyapunov**:
  - **Descripción**: Los exponentes de Lyapunov están superpuestos a la serie temporal de distancias.
  - **Observaciones**: Predominan los exponentes negativos.
  - **Interpretación**: La presencia de exponentes negativos refuerza la idea de estabilidad local, con un comportamiento más regular y menos caótico.
- **Dimensión de Kaplan-Yorke**:
  - **Valor**: 0.0
  - **Interpretación**: Este valor indica una falta de complejidad en la dinámica del sistema, sugiriendo un comportamiento más regular y predecible.

1.9.10 Serie Temporal y Espectro de Exponentes de Lyapunov - Lorentz (choques)

- **Serie Temporal**:
  - **Descripción**: La gráfica muestra la evolución de los choques (colisiones) del caminante aleatorio a lo largo del tiempo.
  - **Observaciones**: Los choques se concentran en un rango de aproximadamente 0 a 8, con alta variabilidad y picos regulares.
  - **Interpretación**: Esto indica un comportamiento impredecible y variado en términos de choques.
- **Exponentes de Lyapunov**:
  - **Descripción**: Los exponentes de Lyapunov están superpuestos a la serie temporal de choques.
  - **Observaciones**: Predominan los exponentes negativos.
  - **Interpretación**: La presencia predominante de exponentes negativos indica estabilidad local en muchas regiones, aunque con comportamiento caótico.
- **Dimensión de Kaplan-Yorke**:
  - **Valor**: 0.0
  - **Interpretación**: Este valor sugiere una menor complejidad en la dinámica del sistema comparado con las gráficas de direcciones, aunque todavía presenta cierta estructura fractal y comportamiento caótico.

distanciaLSTM_caminanteFeigenbaumExponencial

June 24, 2024

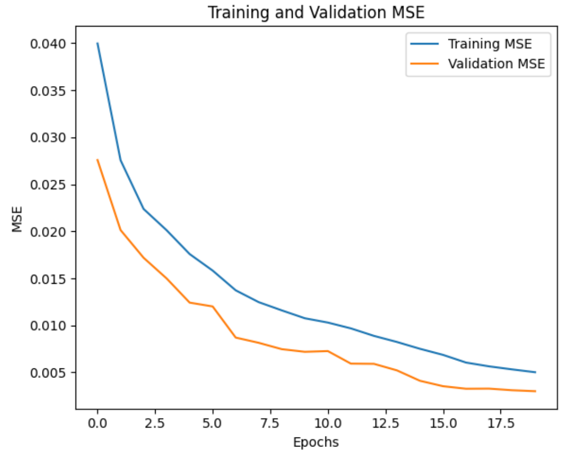
1 PREDICCION: Direccion Feigenbaum Exponencial

[ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout
from tensorflow.keras.metrics import MeanSquaredError

def create_dataset(X, y, time_steps=1):
    Xs, ys = [], []
    for i in range(len(X) - time_steps):
        v = X.iloc[i:(i + time_steps)].values
        Xs.append(v)
        ys.append(y.iloc[i + time_steps])
    return np.array(Xs), np.array(ys)

data = pd.read_csv('salida.csv')

scaler = MinMaxScaler()
data_scaled = scaler.fit_transform(data)

time_steps = 5
X, y = create_dataset(pd.DataFrame(data_scaled), pd.DataFrame(data_scaled),␣
 ↪time_steps)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
 ↪random_state=0)

model = Sequential([
    LSTM(100, activation='relu', input_shape=(X_train.shape[1], X_train.
 ↪shape[2]), return_sequences=True),
    Dropout(0.2),
    LSTM(50, activation='relu'),
    Dropout(0.2),
```

```
    Dense(2)
])

# Compilar el modelo
model.compile(optimizer='adam', loss='mean_squared_error',␣
↪metrics=[MeanSquaredError()])

# Entrenar el modelo
history = model.fit(X_train, y_train, epochs=20, batch_size=32,␣
↪validation_split=0.2)

plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
plt.plot(history.history['loss'], label='Training Loss')
plt.plot(history.history['val_loss'], label='Validation Loss')
plt.title('Training and Validation Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()

plt.subplot(1, 2, 2)
plt.plot(history.history['mean_squared_error'], label='Training MSE')
plt.plot(history.history['val_mean_squared_error'], label='Validation MSE')
plt.title('Training and Validation MSE')
plt.xlabel('Epochs')
plt.ylabel('MSE')
plt.legend()

plt.tight_layout()
plt.show()
```

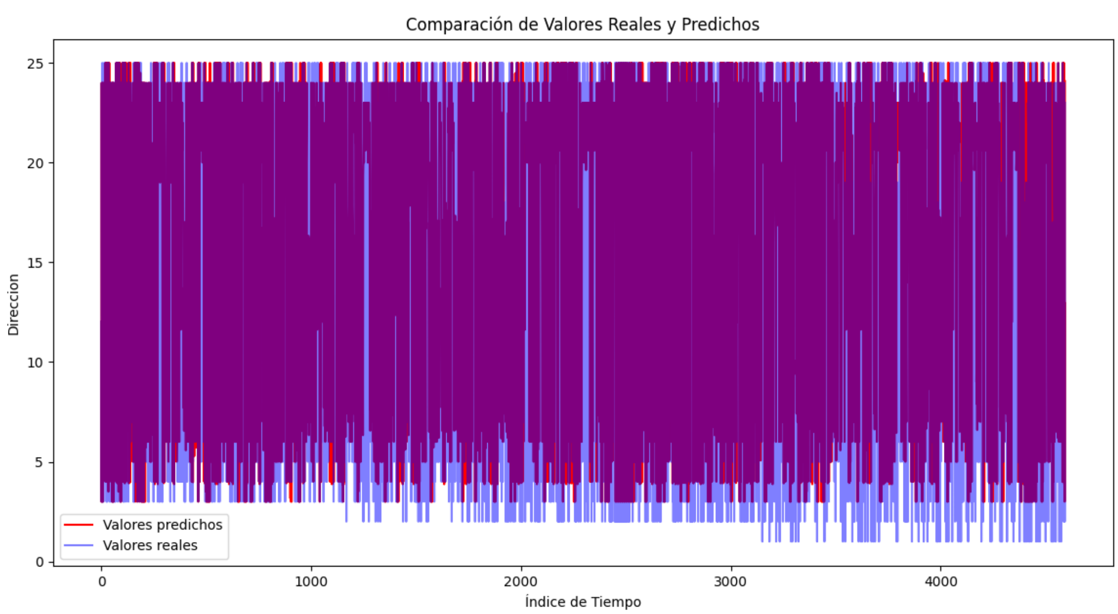
```
Epoch 1/20
460/460 [==============================] - 10s 14ms/step - loss: 0.0400 -
mean_squared_error: 0.0400 - val_loss: 0.0276 - val_mean_squared_error: 0.0276
Epoch 2/20
460/460 [==============================] - 7s 15ms/step - loss: 0.0276 -
mean_squared_error: 0.0276 - val_loss: 0.0201 - val_mean_squared_error: 0.0201
Epoch 3/20
460/460 [==============================] - 6s 12ms/step - loss: 0.0224 -
mean_squared_error: 0.0224 - val_loss: 0.0172 - val_mean_squared_error: 0.0172
Epoch 4/20
460/460 [==============================] - 7s 16ms/step - loss: 0.0201 -
mean_squared_error: 0.0201 - val_loss: 0.0150 - val_mean_squared_error: 0.0150
Epoch 5/20
460/460 [==============================] - 6s 13ms/step - loss: 0.0176 -
mean_squared_error: 0.0176 - val_loss: 0.0124 - val_mean_squared_error: 0.0124
Epoch 6/20
```

```
460/460 [==============================] - 7s 15ms/step - loss: 0.0158 -
mean_squared_error: 0.0158 - val_loss: 0.0120 - val_mean_squared_error: 0.0120
Epoch 7/20
460/460 [==============================] - 6s 13ms/step - loss: 0.0137 -
mean_squared_error: 0.0137 - val_loss: 0.0087 - val_mean_squared_error: 0.0087
Epoch 8/20
460/460 [==============================] - 7s 15ms/step - loss: 0.0125 -
mean_squared_error: 0.0125 - val_loss: 0.0081 - val_mean_squared_error: 0.0081
Epoch 9/20
460/460 [==============================] - 6s 14ms/step - loss: 0.0116 -
mean_squared_error: 0.0116 - val_loss: 0.0075 - val_mean_squared_error: 0.0075
Epoch 10/20
460/460 [==============================] - 6s 12ms/step - loss: 0.0108 -
mean_squared_error: 0.0108 - val_loss: 0.0072 - val_mean_squared_error: 0.0072
Epoch 11/20
460/460 [==============================] - 7s 15ms/step - loss: 0.0103 -
mean_squared_error: 0.0103 - val_loss: 0.0073 - val_mean_squared_error: 0.0073
Epoch 12/20
460/460 [==============================] - 5s 12ms/step - loss: 0.0097 -
mean_squared_error: 0.0097 - val_loss: 0.0059 - val_mean_squared_error: 0.0059
Epoch 13/20
460/460 [==============================] - 7s 14ms/step - loss: 0.0089 -
mean_squared_error: 0.0089 - val_loss: 0.0059 - val_mean_squared_error: 0.0059
Epoch 14/20
460/460 [==============================] - 5s 12ms/step - loss: 0.0082 -
mean_squared_error: 0.0082 - val_loss: 0.0052 - val_mean_squared_error: 0.0052
Epoch 15/20
460/460 [==============================] - 9s 19ms/step - loss: 0.0075 -
mean_squared_error: 0.0075 - val_loss: 0.0041 - val_mean_squared_error: 0.0041
Epoch 16/20
460/460 [==============================] - 5s 11ms/step - loss: 0.0069 -
mean_squared_error: 0.0069 - val_loss: 0.0035 - val_mean_squared_error: 0.0035
Epoch 17/20
460/460 [==============================] - 8s 18ms/step - loss: 0.0061 -
mean_squared_error: 0.0061 - val_loss: 0.0033 - val_mean_squared_error: 0.0033
Epoch 18/20
460/460 [==============================] - 5s 11ms/step - loss: 0.0056 -
mean_squared_error: 0.0056 - val_loss: 0.0033 - val_mean_squared_error: 0.0033
Epoch 19/20
460/460 [==============================] - 6s 12ms/step - loss: 0.0053 -
mean_squared_error: 0.0053 - val_loss: 0.0031 - val_mean_squared_error: 0.0031
Epoch 20/20
460/460 [==============================] - 6s 13ms/step - loss: 0.0050 -
mean_squared_error: 0.0050 - val_loss: 0.0030 - val_mean_squared_error: 0.0030
```



[ ]:
```
print(errorr2)

plt.figure(figsize=(14, 7))
plt.plot(predicted_values, label='Valores predichos', color='red')
plt.plot(actual_values, label='Valores reales', color='blue', alpha=0.5)
plt.title('Comparación de Valores Reales y Predichos')
plt.xlabel('Índice de Tiempo')
plt.ylabel('Direccion')
plt.legend()
plt.show()
```

```
R2 = 0.5804428397112509
```



LSTM_FEIGCUBICA

June 24, 2024

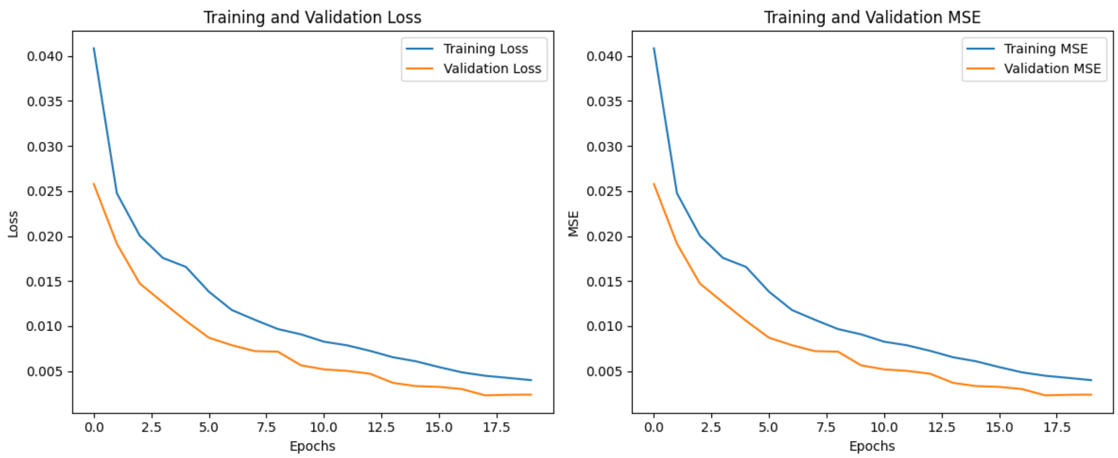
1 PREDICCIÓN Valores X Feigenbaum Exponencial

[ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout
from tensorflow.keras.metrics import MeanSquaredError

def create_dataset(X, y, time_steps=1):
    Xs, ys = [], []
    for i in range(len(X) - time_steps):
        v = X.iloc[i:(i + time_steps)].values
        Xs.append(v)
        ys.append(y.iloc[i + time_steps])
    return np.array(Xs), np.array(ys)

data = pd.read_csv('datosFeigenbaumExponencial.csv')

scaler = MinMaxScaler()
data_scaled = scaler.fit_transform(data)

time_steps = 3
X, y = create_dataset(pd.DataFrame(data_scaled), pd.DataFrame(data_scaled),␣
 ↪time_steps)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
 ↪random_state=0)

model = Sequential([
    LSTM(100, activation='relu', input_shape=(X_train.shape[1], X_train.
 ↪shape[2]), return_sequences=True),
    Dropout(0.2),
    LSTM(50, activation='relu'),
    Dropout(0.2),
```

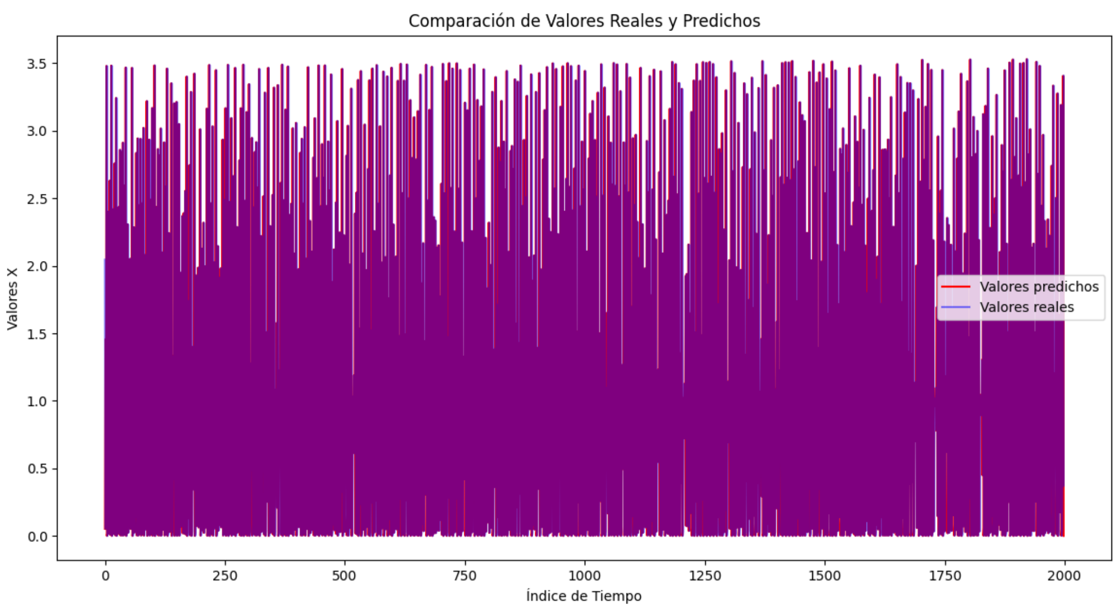
```
    Dense(2)
])

model.compile(optimizer='adam', loss='mean_squared_error',␣
 ↪metrics=[MeanSquaredError()])

history = model.fit(X_train, y_train, epochs=20, batch_size=32,␣
 ↪validation_split=0.2)

plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
plt.plot(history.history['loss'], label='Training Loss')
plt.plot(history.history['val_loss'], label='Validation Loss')
plt.title('Training and Validation Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()

plt.subplot(1, 2, 2)
plt.plot(history.history['mean_squared_error'], label='Training MSE')
plt.plot(history.history['val_mean_squared_error'], label='Validation MSE')
plt.title('Training and Validation MSE')
plt.xlabel('Epochs')
plt.ylabel('MSE')
plt.legend()

plt.tight_layout()
plt.show()
```

```
Epoch 1/20
460/460 [==============================] - 9s 10ms/step - loss: 0.0409 -
mean_squared_error: 0.0409 - val_loss: 0.0258 - val_mean_squared_error: 0.0258
Epoch 2/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0248 -
mean_squared_error: 0.0248 - val_loss: 0.0191 - val_mean_squared_error: 0.0191
Epoch 3/20
460/460 [==============================] - 6s 12ms/step - loss: 0.0200 -
mean_squared_error: 0.0200 - val_loss: 0.0147 - val_mean_squared_error: 0.0147
Epoch 4/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0176 -
mean_squared_error: 0.0176 - val_loss: 0.0126 - val_mean_squared_error: 0.0126
Epoch 5/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0166 -
mean_squared_error: 0.0166 - val_loss: 0.0106 - val_mean_squared_error: 0.0106
Epoch 6/20
460/460 [==============================] - 6s 12ms/step - loss: 0.0138 -
mean_squared_error: 0.0138 - val_loss: 0.0087 - val_mean_squared_error: 0.0087
```

```
Epoch 7/20
460/460 [==============================] - 4s 10ms/step - loss: 0.0118 -
mean_squared_error: 0.0118 - val_loss: 0.0079 - val_mean_squared_error: 0.0079
Epoch 8/20
460/460 [==============================] - 5s 11ms/step - loss: 0.0107 -
mean_squared_error: 0.0107 - val_loss: 0.0072 - val_mean_squared_error: 0.0072
Epoch 9/20
460/460 [==============================] - 6s 13ms/step - loss: 0.0097 -
mean_squared_error: 0.0097 - val_loss: 0.0071 - val_mean_squared_error: 0.0071
Epoch 10/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0091 -
mean_squared_error: 0.0091 - val_loss: 0.0056 - val_mean_squared_error: 0.0056
Epoch 11/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0083 -
mean_squared_error: 0.0083 - val_loss: 0.0052 - val_mean_squared_error: 0.0052
Epoch 12/20
460/460 [==============================] - 5s 11ms/step - loss: 0.0079 -
mean_squared_error: 0.0079 - val_loss: 0.0050 - val_mean_squared_error: 0.0050
Epoch 13/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0072 -
mean_squared_error: 0.0072 - val_loss: 0.0047 - val_mean_squared_error: 0.0047
Epoch 14/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0065 -
mean_squared_error: 0.0065 - val_loss: 0.0037 - val_mean_squared_error: 0.0037
Epoch 15/20
460/460 [==============================] - 6s 12ms/step - loss: 0.0061 -
mean_squared_error: 0.0061 - val_loss: 0.0033 - val_mean_squared_error: 0.0033
Epoch 16/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0054 -
mean_squared_error: 0.0054 - val_loss: 0.0032 - val_mean_squared_error: 0.0032
Epoch 17/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0048 -
mean_squared_error: 0.0048 - val_loss: 0.0030 - val_mean_squared_error: 0.0030
Epoch 18/20
460/460 [==============================] - 6s 12ms/step - loss: 0.0045 -
mean_squared_error: 0.0045 - val_loss: 0.0023 - val_mean_squared_error: 0.0023
Epoch 19/20
460/460 [==============================] - 4s 9ms/step - loss: 0.0042 -
mean_squared_error: 0.0042 - val_loss: 0.0024 - val_mean_squared_error: 0.0024
Epoch 20/20
460/460 [==============================] - 5s 10ms/step - loss: 0.0040 -
mean_squared_error: 0.0040 - val_loss: 0.0024 - val_mean_squared_error: 0.0024
```



[ ]:
```
plt.figure(figsize=(14, 7))
plt.plot(predicted_values, label='Valores predichos', color='red')
plt.plot(actual_values, label='Valores reales', color='blue', alpha=0.5)
plt.title('Comparación de Valores Reales y Predichos')
plt.xlabel('Índice de Tiempo')
plt.ylabel('Valores X')
plt.legend()
plt.show()
```



2 ANEXOS

Los códigos utilizados se pueden encontrar en cualquiera de estos enlaces:

Google Drive: https://drive.google.com/drive/folders/1q-YwDGhG-700djaiBptVDcwc_OYP1dE0?usp=sharing

GitHub: https://github.com/RodrigoG13/Time-Series.git