



Universidad Autónoma Metropolitana
Unidad Iztapalapa
División de Ciencias Básicas e Ingeniería
Posgrado en Física

**“Análisis y estudio de sistemas dinámicos con
herramientas de machine learning ”**

Tesis que presenta

Fidel Montoya Molina
Matrícula: 2222800171
Correo: f.montoyam9@gmail.com

Para obtener el grado de **Maestro en Ciencias (Física)**

Director de Tesis: Dr. Adrián Mauricio Escobar Ruiz

Jurado:

Presidente: Dr. José Luis Del Río Correa
Secretario: Dr. Adrián Mauricio Escobar Ruiz
Vocal: Dr. Mario Alan Quiroz Juárez

Iztapalapa, CDMX, 31 de Mayo 2024.

Dedicado a mis padres Jaciel y Verónica.

*”...Más vale una vida sencilla pero
tranquila...”*

Agradecimientos

- A mis padres Jaciel Montoya y Verónica Molina por su paciencia, comprensión, apoyo y amor a lo largo de todos estos años; sin su guía y consejos nada de esto sería posible. Por enseñarme lo valioso de la educación y los valores en esta vida.
- A mi hermana Xóchitl por ser mi motivación para mejorar día con día tanto en lo profesional como lo personal, y a mi hermano Jaciel por ser un pilar en mi amor por la música, fundamental para esas noches de largo estudio y trabajo.
- A mis profesores: el Dr. Adrián Escobar por su acompañamiento académico estos últimos años de mi formación profesional y el siempre motivarme en el pensamiento crítico y científico; el Dr. José Luis del Río Correa por brindarme un espacio en el cual poder estudiar en tranquilidad.
- Agradezco también al Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCyT) por la beca otorgada durante este periodo con número de CVU: 1235760 y el apoyo otorgado a través del Proyecto de Ciencia de Frontera CF-2023-I-1496.
- A mis amigos: De toda la vida, Gerardo, Aldo y María por tantas experiencias y enseñanzas. A Jason, Abraham, Misael, Alejandro y César por su acompañamiento en esta etapa académica en la cual todos nos apoyamos y aprendimos juntos.
- A Paty Lecuona, por acompañarme en esta etapa académica y enseñarme lo valioso que tiene cada momento de la vida.

Índice general

Índice general	I
1. Resumen	1
2. Abstract	3
3. Introducción	5
4. Sistemas Dinámicos	9
4.1. Sistemas dinámicos	9
4.1.1. Sistemas discretos	9
4.1.2. Sistemas continuos	10
4.2. Espacio fase	11
4.2.1. Sistemas lineales $n = 2$	13
4.3. Sistemas Hamiltonianos	15
4.3.1. Sección de Poincaré	17
4.3.2. Líneas de Simetría	17
4.4. Métodos de integración numérica	20
4.4.1. Método de Euler	21
4.4.2. Método de Runge-Kutta de orden 2 y orden 4	23
5. Regresión	27

5.1. Regresión Lineal	27
5.2. Mínimos Cuadrados	28
5.2.1. Caso simple	28
5.2.2. Ejemplo en Python: regresión con una variable independiente . .	31
5.2.3. Caso multivariable	31
5.2.4. Coeficiente de correlación	35
5.2.5. Ejemplo caso multivariable (β_1, β_2)	36
5.3. Regularización	37
5.3.1. Optimización	41
6. Algoritmo de aprendizaje automático: SINDy	45
6.1. SINDy	45
6.2. Paquetería PySINDy	49
6.3. Reconstrucción de ecuaciones dinámicas con PySINDy: Ejemplos	50
6.3.1. Oscilador lineal y cúbico	51
6.3.2. Péndulo Simple	52
6.4. Atractor de Lorenz	56
7. Sistema generalizado de Hénon-Heiles	61
7.1. Sistema de Hénon-Heiles	61
7.1.1. Potencial de Hénon Heiles	62
7.1.2. Ecuaciones de movimiento	64
7.2. Sistema Generalizado de Hénon Heiles	65
7.2.1. Sección de Poincaré y exponentes de Lyapunov	68
7.2.2. Líneas de simetría	70
7.3. Reconstrucción de las ecuaciones de Hamilton con SINDy	72
7.3.1. Resultados	79
8. Modelo de ECG como sistema dinámico	91
8.1. Generación de un electrocardiograma (ECG)	91
8.2. Reconstrucción de ecuaciones dinámicas con SINDy	95
8.3. Construcción de modelos a partir de una variable de estado	98
8.3.1. Teorema de Takens	99
8.3.2. Atractor de Lorenz mediante el método de retardo temporal . .	100

9. Conclusiones	107
Bibliografía	111
A. Apéndice A	115
A.1. Código en Python: Integración numérica	115
B. Apéndice B	119
B.1. Demostración para la solución multivariable de mínimos cuadrados . .	119
B.2. Código en Python: Regresión lineal univariable	120
B.3. Programa en Python: Regresión lineal multivariable	122
C. Apéndice C	125
C.1. Código en Python: Regularización STLQ	125
C.2. Código en Python: Frente de Pareto	128
D. Apéndice D	131
D.1. Cálculo de condiciones iniciales aleatorias que conservan energía	131
E. Apéndice E	135
E.1. Código en Python: Reconstrucción de ecuaciones con SINDy	135
F. Apéndice F	141

En este trabajo de tesis se investiga la precisión de algoritmos de aprendizaje automático (Machine Learning) para reconstruir modelos de sistemas dinámicos que presentan caos. En particular, nos enfocaremos en una generalización del célebre sistema Hamiltoniano de Hénon-Heiles. Finalmente y de forma breve se estudia la aplicación de estos algoritmos en la predicción de arritmias en el corazón. Se plantean también algunas ideas para trabajos posteriores sobre la implementación de este tipo de algoritmos en el caso donde no se tiene acceso a la totalidad de datos y la información del modelo es incompleta. El contenido se estructura de la siguiente manera: la Introducción se presenta en el Capítulo 1, el marco teórico se describe en los Capítulos 2-4, los resultados se muestran en los Capítulos 5-6 y, finalmente, se exponen las Conclusiones en el Capítulo 7. De forma más detallada:

- En el Capítulo 2, se explican brevemente los elementos básicos de la teoría de sistemas dinámicos. Se incluyen los conceptos tales como el espacio fase y las herramientas usuales que se utilizan en el análisis de la dinámica del sistema (secciones de Poincaré, líneas de simetría).
- En el Capítulo 3, se discute la teoría de regresión lineal por mínimos cuadrados, que es parte del aprendizaje automático (Machine Learning), la cual permite construir una relación entre variables dependientes e independientes del conjunto de datos disponibles, reconociendo patrones y obteniendo un modelo matemático que describa la dinámica. Es importante enfatizar que se buscan modelos parsimonios, es decir que sean precisos (en términos del valor del error cuadrático medio y el coeficiente de correlación correspondiente) y sencillos (que sea representado por el

menor número de términos en las ecuaciones de dinámicas), para esto se emplean herramientas del aprendizaje automático como la regularización y optimización, técnicas que también se abordan en este Capítulo.

- En el Capítulo 4, se presenta un algoritmo de aprendizaje automático (Machine Learning) que (re)construye las ecuaciones gobernantes de sistemas no lineales, denominado SINDy (Sparse Identification of Nonlinear Dynamics). Está basado en la aplicación de regresión lineal en conjunto con técnicas de regularización y optimización, en las series de tiempo (problema inverso) recopiladas por los experimentos o simulaciones.
- Con SINDy, en el Capítulo 5 se realiza la reconstrucción de un modelo caótico inspirado en el célebre sistema de Hénon-Heiles. El principal objetivo es estudiar cualitativa y cuantitativamente la precisión del algoritmo SINDy como función de la energía del sistema y los tipos de trayectorias que se presentan. En este modelo coexisten regiones de regularidad y caos.
- En el Capítulo 6 se investiga brevemente la reconstrucción de las ecuaciones gobernantes para un modelo que genera señales de electrocardiogramas (ECG) en el que se presentan transiciones entre dinámicas periódicas y caóticas.
- Finalmente, en el Capítulo 7 se resumen los resultados de este trabajo de Tesis, se presentan las conclusiones y se plantean temas de investigación para desarrollar en trabajos futuros.

Como resultado de este trabajo de tesis se realizaron las siguientes actividades: Elaboración de un artículo de investigación, publicado en la revista *Chaos, Solitons & Fractals* bajo el título *Data-driven reconstruction of chaotic dynamical equations: the Hénon-Heiles type system* (<https://www.sciencedirect.com>) ; un artículo de divulgación para la Gaceta UNAM (<https://www.campusjuriquilla.unam.mx>); participación con ponencias en el LXVI Congreso Nacional de Física organizado por la Sociedad Mexicana de Física y en el Congreso Nacional de Divulgación de la Ciencia y la Técnica organizado por Somedicyt en el contexto de esta tesis. Como actividad extra, se elaboró un artículo de investigación publicado en la revista *Journal of Physics B: Atomic, Molecular and Optical Physics* titulado *Three-body harmonic molecule* (<https://iopscience.iop.org/article/10.1088/1361-6455/acbdf0>).

In this thesis work, the accuracy of modern machine learning algorithms is investigated for reconstructing models of chaotic dynamical systems. In particular, we will focus on a generalization of the famous Hamiltonian system of Hénon-Heiles. Finally, a brief study is conducted on the application of these algorithms in predicting arrhythmias in the heart. Some ideas for future work are also proposed regarding the implementation of such algorithms in cases where access to the entirety of data is not available and model information is incomplete.

The content is structured as follows: the Introduction is presented in Chapter 1, the theoretical framework is described in Chapters 2-4, the results are shown in Chapters 5-6, and finally, the Conclusions are presented in Chapter 7. In more detail:

- Chapter 2 briefly explains the basic elements of dynamical systems theory. Concepts such as phase space and the usual tools used in system dynamics analysis (Poincaré surface sections, symmetry lines) are included.
- In Chapter 3, the theory of linear regression using least squares is discussed, which it's an area of Machine Learning. This method allows to build a relationship between dependent and independent variables from the available data set, identifying patterns and creating a mathematical model that describes the dynamics. It is important to emphasize that parsimonious models are sought, meaning that they should be accurate (in terms of the mean squared error and the corresponding correlation coefficient) and simple (represented by the fewest possible terms in the dynamic equations). Machine Learning tools like regularization and optimization are used to achieve this, techniques that are also covered in this chapter.

- Chapter 4 presents a machine learning algorithm that (re)constructs the governing equations of nonlinear systems called SINDy (Sparse Identification of Nonlinear Dynamics) . It is based on the application of linear regression together with regularization and optimization techniques, on time series (inverse problem) collected from experiments or simulations.
- Using SINDy, Chapter 5 performs the reconstruction of a chaotic model inspired by the famous Hénon-Heiles system. The main objective is to qualitatively and quantitatively study the accuracy of the SINDy algorithm as a function of system energy and model parameters. This model exhibits regions of regularity and chaos.
- Chapter 6 briefly investigates the reconstruction of governing equations for a model that generates electrocardiogram (ECG) signals with transitions between periodic and chaotic dynamics.
- Finally, Chapter 7 summarizes the results of this thesis work, presents the conclusions, and proposes research topics for future work.

As a result of this thesis work, the following activities were carried out: A published research article for the journal Chaos, Solitons & Fractals titled Data-driven reconstruction of chaotic dynamical equations: the Hénon-Heiles type system (<https://www.sciencedirect.com/>); a dissemination article for Gaceta UNAM (<https://www.campusjuriquilla.unam.mx>); and participation with presentations at the LXVI National Physics Congress organized by Mexican Physics Society and in the National Congress for the Dissemination of Science and technique organized by Somedicyt in the context of this thesis.

As an additional activity, a research article was prepared and published in the journal Journal of Physics B: Atomic, Molecular and Optical Physics titled Three-body harmonic molecule (<https://iopscience.iop.org/article/10.1088/1361-6455/acbdf0>).

Introducción

“... If nature were not beautiful, it would not be worth knowing, and if nature were not worth knowing, life would not be worth living.”

Henri Poincaré

Durante los últimos siglos, desde que Isaac Newton aplicó ecuaciones para describir sistemas físicos que evolucionan en el tiempo, se han alcanzado avances teóricos significativos para validar o prever el comportamiento de estos sistemas en diversas áreas científicas. Sin embargo, la comprensión y predicción precisa de sistemas dinámicos ha sido un desafío constante debido a la complejidad inherente de muchos fenómenos naturales. Para la ciencia es fundamental la suposición de que los experimentos son predecibles y repetibles. Por lo tanto, resultó sorprendente para la mayoría de los científicos encontrar sistemas deterministas simples que no cumplieran con estas expectativas. En cambio, demostraron características “caóticas”, donde incluso la mínima alteración en las condiciones iniciales producía resultados muy diferentes, aun cuando las ecuaciones que rigen el sistema fueran conocidas con exactitud.

En 1887 la teoría de sistemas dinámicos experimentó un cambio radical cuando el Rey Oscar II propuso un premio para quien resolviera el problema de los n cuerpos, con el objetivo de demostrar la estabilidad del sistema solar. Aunque Newton había resuelto el problema de dos cuerpos geoméricamente, fue Henri Poincaré quien, en 1889, recibió el premio, no por hallar una solución, sino por demostrar su inexistencia. En su ensayo de 1903, *Ciencia y Método*, Poincaré discutió la predicción de eventos

y cómo la dinámica varía con el tiempo debido a las condiciones iniciales [1]. Fue así como Poincaré introdujo la noción de caos en sistemas deterministas, indicando que el futuro de cualquier sistema está definido por su pasado. El caos, caracterizado por la no repetitividad, sensibilidad a condiciones iniciales, y dependencia de parámetros del sistema, suele aparecer en sistemas regidos por ecuaciones no lineales y ha sido objeto de investigación durante mucho tiempo.

Sería fascinante lograr una predicción sin errores sobre eventos futuros a través de ecuaciones, sin importar las condiciones iniciales de la dinámica y el tipo de trayectoria. Esta habilidad sería invaluable para comprender y anticipar una amplia gama de fenómenos naturales o procesos complejos, para resolver y entender a fondo problemas de la cotidianidad como la situación sanitaria en 2019 donde predicciones médicas exactas hubiesen sido clave para el manejo de la pandemia; otros ejemplos de aplicación podrían ser el pronóstico de desastres naturales, condiciones meteorológicas o predicción del tráfico en las ciudades.

Esta búsqueda, digamos de perfeccionismo, nos conduce a un territorio emocionante: el aprendizaje automático (Machine Learning). Los orígenes de este campo están íntimamente ligados al estudio de sistemas complejos y la necesidad de descubrir patrones en datos dinámicos, a menudo caóticos y aprender de ellos. Los orígenes del aprendizaje automático se remontan a la propuesta de Warren McCulloch y Walter Pitts en la descripción de una neurona artificial y su relación con el cerebro humano produce patrones complejos [2]. A partir de fundamentos matemáticos, los algoritmos de aprendizaje automático, se enfocan en la predicción y el reconocimiento de patrones dentro de grandes conjuntos de datos, ofreciendo un nuevo paradigma para entender y manejar la complejidad del mundo que nos rodea. En este sentido, seguimos el legado de visionarios como Poincaré, cuyas ideas continúan inspirando la exploración de los límites del conocimiento y la innovación en la ciencia.

En la actualidad, se han desarrollado diversos tipos de algoritmos de aprendizaje automático para abordar una amplia gama de problemas de diversa complejidad. Entre estos algoritmos se encuentran el Dynamic Mode Decomposition (DMD), la teoría del operador de Koopman y la Sparse Identification of Nonlinear Dynamics (SINDy). Estas herramientas computacionales se emplean para aproximar soluciones a problemáticas multidisciplinares.

En particular, se hace énfasis en una revisión sistemática en el presente trabajo de

tesis hacia el algoritmo SINDy. Se persigue su aplicación en la construcción de modelos de ecuaciones que describan de manera precisa las dinámicas no lineales. El objetivo principal radica en evaluar la capacidad del algoritmo para capturar y representar adecuadamente la complejidad inherente a diferentes sistemas, así como medir su precisión en la predicción de comportamientos caóticos en diversos contextos. Para esto se utiliza como sistema de prueba, una generalización del sistema de Hénon-Heiles (el cual originalmente fue propuesto por astrónomos en la descripción de las trayectorias de una estrella alrededor de una galaxia axisimétrica) para dinámicas tanto periódicas o que presentan caos y particulares para ciertos valores de la energía mecánica, obteniendo así las ecuaciones de Hamilton ya conocidas a partir de series de tiempo calculadas numéricamente. Sobre esta parte del trabajo de tesis se encontraron resultados de los cuales se realizó un artículo de investigación, el cual está publicado en la revista *Chaos, Solitons & Fractals*.

Como complemento y para reafirmar algunos resultados del primer sistema comentado anteriormente, se presenta brevemente la aplicación del algoritmo SINDy para reconstruir las ecuaciones de un modelo que genera señales de electrocardiogramas; estas señales pueden generar dinámicas caóticas cuando se realiza una variación de un parámetro de control en las ecuaciones dinámicas, que están ampliamente ligadas a condiciones como arritmias en el corazón las cuales pueden ser mortales si no se realiza un seguimiento médico pertinente. Con los resultados obtenidos de ambos sistemas se plantean algunas interrogantes que pueden abrir líneas de investigación para trabajos futuros: Utilizando técnicas de aprendizaje automático, ¿Es posible mitigar la pérdida de precisión a tiempos largos para sistemas que presentan caos?, y en caso de un sistema dinámico donde solo se pueda tener acceso a la medición de una sola variable dinámica, usando algún tipo de algoritmo, ¿Se pueden construir ecuaciones de movimiento y replicar la dinámica a pesar del caos?; para esto último se aborda una idea introductoria para su desarrollo en la última parte del trabajo.

Sistemas Dinámicos

En este primer Capítulo se examinan los principios subyacentes para describir la evolución de sistemas dinámicos y Hamiltonianos en el tiempo. Se realiza una breve descripción de herramientas como la sección de Poincaré útil para estudiar la estabilidad de los puntos fijos (de equilibrio) en el espacio fase sin necesidad de una solución analítica y líneas de simetría que ayudan a encontrar puntos en la sección que corresponden a órbitas periódicas. Por último, de forma general se describen algunos métodos para realizar integración numérica en ecuaciones de movimiento.

4.1. Sistemas dinámicos

Se define a un sistema dinámico como aquel que evoluciona en el tiempo t y posee n variables de estado x_i , $i = 1, 2, 3, \dots, n$, las cuales contienen la información completa para la descripción del sistema [3]. Considerando que la evolución temporal se puede realizar en pasos de tiempo finitos ó infinitesimales, los sistemas se clasifican en discretos y continuos, respectivamente; estos se describen a continuación.

4.1.1. Sistemas discretos

Considerando el caso $n = 1$, donde sólo se tiene una variable de estado $x(t)$ que adquiere valores discretos $x_k = x(t_k)$ de los cuales $x_{k+1} = x(t_{k+1})$ definen la evolución temporal del sistema. Se define $\mathcal{F}$ como la evolución en forma de mapeo, donde $x_{k+1} = \mathcal{F}(x_k)$ y $t_{k+1} = t_k + \Delta t$. A partir del punto donde comienza a evolucionar el sistema (condición inicial) x_0 , se genera una órbita definida por el conjunto

$x, \mathcal{F}(x), \mathcal{F}(x)^2, \dots, \text{etc.}$

Entonces dado $x_0 \in \mathbb{R}$, la órbita generada por x_0 es el conjunto de puntos:

$$x_0, x_1 = \mathcal{F}(x_0), x_2 = \mathcal{F}(\mathcal{F}(x_0)), \dots, x_n = \mathcal{F}(x_{n-1}), \quad (4.1)$$

o bien

$$x_0, x_1 = \mathcal{F}(x_0), x_2 = \mathcal{F}(x_0)^2, \dots, x_n = \mathcal{F}(x_0)^n, \quad (4.2)$$

lo cual es la iteración de x_0 bajo el mapeo $\mathcal{F}$. Lo anterior se puede generalizar para n variables de estado.

4.1.2. Sistemas continuos

Considerando un sistema de orden n , cuando $\Delta t \rightarrow 0$, se considera el sistema como continuo, es decir, la separación entre cada paso temporal es infinitesimal y se obtiene una ecuación diferencial. Ahora la razón de cambio temporal de cualquier variable de estado se escribe como:

$$\frac{dx_i}{dt} = \dot{x}_i \equiv \vec{f}_i(x_i, t) \quad , \quad i = 1, 2, 3, \dots, n, \quad (4.3)$$

donde $x_i = x_i(t)$, y es llamada ecuación de movimiento. La ecuación diferencial tiene soluciones que son funciones temporales implícitas (sistemas autónomos) o explícitas del tiempo (no-autónomos). Introduciendo $\vec{x} = (x_1, x_2, \dots, x_n)$, el sistema autónomo se conforma por n ecuaciones

$$\begin{aligned} \dot{x}_1 &= \vec{f}_1(\vec{x}), \\ \dot{x}_2 &= \vec{f}_2(\vec{x}), \\ &\vdots \\ \dot{x}_n &= \vec{f}_n(\vec{x}), \end{aligned}$$

de forma compacta se escribe como

$$\dot{\vec{x}} = \vec{f}(\vec{x}(t)). \quad (4.4)$$

Los puntos $\vec{x}(t)$ donde la velocidad fase se anula, $\vec{f}(\vec{x}, t) = 0$, se denominan como *puntos fijos* $\vec{x}_p$ y representan puntos de estabilidad o inestabilidad del sistema. Cerca de un punto estable se tiende a permanecer cerca del mismo estado y en un inestable cualquier perturbación saca del equilibrio al sistema. La riqueza del sistema reside en resolver (4.4) para los diferentes valores de los parámetros que contenga el modelo, la naturaleza de los puntos fijos depende de ellos. Las soluciones $\vec{x}(t)$ de (4.4) se definen como las curvas integrales (trayectorias) del sistema.

4.2. Espacio fase

Para un sistema de orden n con variables $x_1, x_2, \dots, x_n$, se escribe el vector $\vec{x} = (x_1, x_2, \dots, x_n)$ que habita en un espacio abstracto n -dimensional denominado *espacio fase* [3]. Para un sistema dinámico, la evolución en este espacio se determina con los diferentes valores de $\vec{x}$ y t en $f(\vec{x}, t)$, lo cual conforma la velocidad fase. Cuando la función velocidad es independiente del tiempo el sistema se determina autónomo. Durante cada trayectoria del sistema se traza una curva continua en el espacio fase llamada curva fase, el conjunto de todas ellas conforman el flujo fase. Para cada condición inicial $\vec{x}(t = 0)$, existe una única solución de la ecuación diferencial (4.4) y por lo tanto una sola curva fase, entre todas ellas nunca existirá una intersección debido al teorema de existencia y unicidad. Una de las ventajas de usar el espacio fase radica en la posibilidad de estudiar de forma cualitativa la dinámica del sistema sin necesidad de una solución analítica.

En particular, para un sistema continuo 2 – dimensional ($n = 2$), el espacio fase es un plano. Usando coordenadas cartesianas, donde $\vec{x}(t) = (x_1(t), x_2(t))$, las ecuaciones diferenciales correspondientes son:

$$\frac{dx_1}{dt} = \dot{x}_1 = f_1(x_1, x_2, t) , \quad (4.5)$$

$$\frac{dx_2}{dt} = \dot{x}_2 = f_2(x_1, x_2, t) , \quad (4.6)$$

que forman un conjunto de ecuaciones acopladas de primer orden. Cuando el sistema es autónomo, f_1, f_2 son independientes de t . Identificando $x_1 = x, f_1 = f_x$ junto con $x_2 = y, f_2 = f_y$ una forma de eliminar la dependencia temporal en (4.5),(4.6) , es

considerar

$$\frac{dy}{dx} = \frac{f_y(x, y)}{f_x(x, y)} . \quad (4.7)$$

Como ejemplo, tomando la ecuación de segundo orden:

$$\ddot{x} + \omega^2 x = 0 , \quad (4.8)$$

que reescribiéndola como un conjunto de ecuaciones de primer orden, tenemos

$$\dot{x} = y , \quad (4.9)$$

$$\dot{y} = -\omega^2 x , \quad (4.10)$$

y se obtiene con ayuda de (4.7):

$$\frac{dy}{dx} = \frac{-\omega^2 x}{y} . \quad (4.11)$$

Integrando esta última ecuación se obtiene una cantidad constante en el tiempo,

$$\frac{y^2}{2} + \frac{\omega^2}{2} x^2 = Cte , \quad (4.12)$$

o bien

$$\frac{\dot{x}^2}{2} + \frac{\omega^2}{2} x^2 = Cte , \quad (4.13)$$

lo cual corresponde a la energía del sistema.

Entonces si existe una constante de movimiento, este se encuentra limitado en el espacio fase; para el oscilador armónico simple, la energía (4.13) es una constante de movimiento y la curva fase es una elipse concéntrica al punto de equilibrio ($x = y = 0$) donde cada una depende del valor de la energía. Por otro lado, en el caso de un oscilador lineal amortiguado, no se tiene una constante de movimiento. Las curvas fase caen de forma espiral al punto de equilibrio, dado por $x = y = 0$.

En general, el conjunto de ecuaciones diferenciales que describen el movimiento de un sistema dinámico debe ser reescrito como $\dot{\vec{x}} = f(\vec{x}, t)$, definiendo las derivadas de

orden superior como las nuevas variables dependientes. El número de ecuaciones diferenciales de primer orden definen el orden del sistema. Para un sistema Newtoniano con n grados de libertad y nulas restricciones se tiene un sistema de orden $2n$ (ecuaciones de Hamilton). Se puede obtener una comprensión detallada del comportamiento del sistema mediante un análisis del flujo fase. En el espacio de fase, cada punto fijo revela la estructura del flujo de fase en su entorno inmediato, y en conjunto, proporcionan una perspectiva total de la naturaleza del flujo en el espacio fase.

4.2.1. Sistemas lineales $n = 2$

En el caso particular de un sistema autónomo de segundo orden $n = 2$, con $\vec{x} = (x, y)$, conociendo la velocidad fase $\vec{f}(x, y) = (f_x, f_y)$, se resuelven de forma simultánea las ecuaciones

$$f_x(x, y) = 0, \quad , \quad f_y(x, y) = 0, \quad (4.14)$$

para determinar la posición de los puntos fijos $\vec{x}_p$ y así obtener una matriz para el flujo linealizado en la vecindad de cada punto fijo. Esto se obtiene realizando una expansión en serie de Taylor de la función velocidad alrededor de $\vec{x}_p$, donde el primer término corresponde a la parte lineal, por lo tanto

$$\begin{bmatrix} f_x(x, y) \\ f_y(x, y) \end{bmatrix} = \begin{bmatrix} \frac{\partial f_x}{\partial x} & \frac{\partial f_x}{\partial y} \\ \frac{\partial f_y}{\partial x} & \frac{\partial f_y}{\partial y} \end{bmatrix} \begin{bmatrix} x - x_p \\ y - y_p \end{bmatrix} + \mathcal{O}(\vec{x} - \vec{x}_k), \quad (4.15)$$

donde las derivadas son evaluadas en $\vec{x}_p$. Entonces alrededor del punto fijo,

$$f(x, y) = A(\vec{x} - \vec{x}_p) + \mathcal{O}(\vec{x} - \vec{x}_p), \quad (4.16)$$

se calculan los eigenvalores de la matriz A (matriz Jacobiana) evaluada en los puntos encontrados, explícitamente

$$A = \begin{bmatrix} \frac{\partial f_x}{\partial x} & \frac{\partial f_x}{\partial y} \\ \frac{\partial f_y}{\partial x} & \frac{\partial f_y}{\partial y} \end{bmatrix}, \quad (4.17)$$

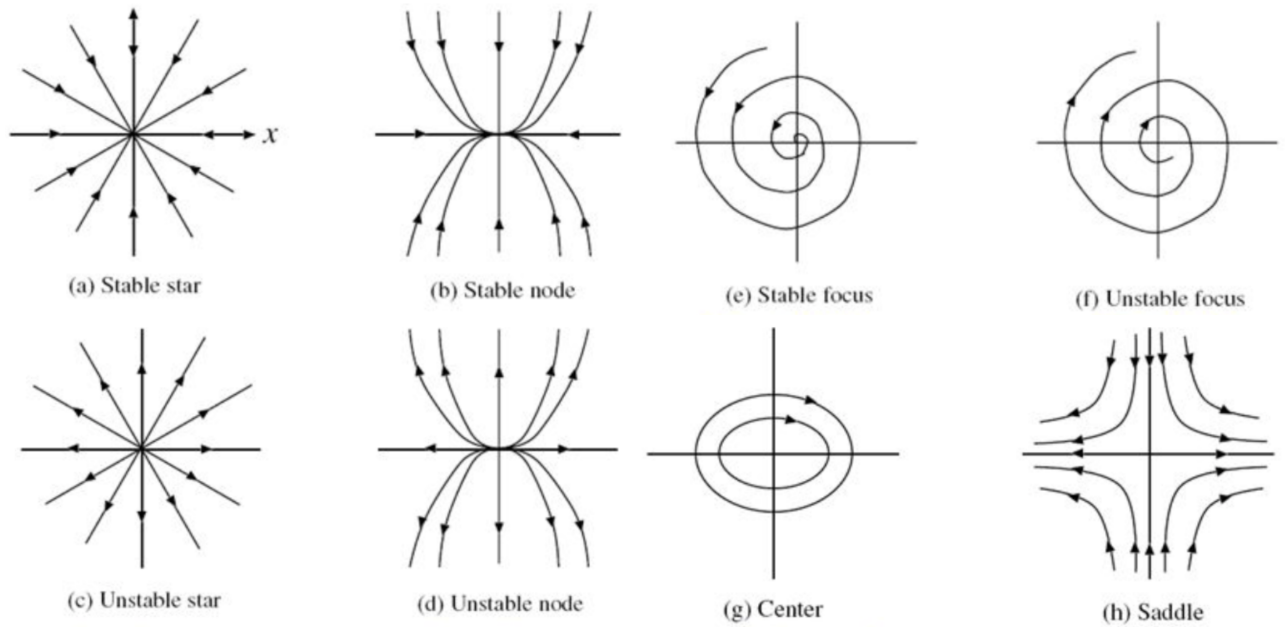


Figura 4.1: Representación gráfica para la clasificación de puntos fijos.

en $\vec{x} = \vec{x}_p$ de la naturaleza de los eigenvalores de A , los puntos fijos se clasifican de la siguiente forma, Fig. 4.1:

- (a) $\lambda_1 = \lambda_2 < 0$, *estrella estable*.
- (b) $\lambda_1 < \lambda_2 < 0$, *nodo estable*.
- (c) $\lambda_1 = \lambda_2 > 0$, *nodo inestable*.
- (d) $\lambda_1 > \lambda_2 > 0$, *nodo inestable*.
- (e) $\lambda_1 = -\alpha + i\beta$, $\lambda_2 = -\alpha - i\beta$, *espiral estable*.
- (f) $\lambda_1 = \lambda_2^* = \alpha + i\beta$, $\alpha > 0$, $\omega < 0$, *espiral inestable*.
- (g) $\lambda_1 = \lambda_2^* = i\omega$, *elíptico central*.
- (h) $\lambda_1 = \lambda_2 < 0$, *punto silla*.

4.3. Sistemas Hamiltonianos

Un caso particular de los sistemas dinámicos son los sistemas Hamiltonianos [4], para los cuales se intercambia en (4.4) la función velocidad por derivadas parciales de una función escalar $\mathcal{H}(q, p, t)$ denominada Hamiltoniano que depende de las $2n$ variables canónicas q_i, p_i ($i = 1, 2, \dots, n$), formada por la energía cinética K y potencial V . Cuando el sistema es autónomo, este no depende explícitamente del tiempo por lo que $\mathcal{H}(q(t), p(t))$, se escribe como

$$\mathcal{H} = K + V = \sum_{i=1}^n \frac{p_i^2}{2m} + V(q_1, q_2, \dots, q_n) . \quad (4.18)$$

Definiendo aquí $\vec{r} = (q_1, q_2, \dots, q_n; p_1, p_2, \dots, p_n)$ denota un punto en el espacio fase. Las ecuaciones diferenciales de primer orden son obtenidas mediante,

$$\dot{\vec{r}}_i = \mathcal{J} \nabla \mathcal{H}(\vec{r}, t) , \quad (4.19)$$

con,

$$\nabla = \left(\frac{\partial}{\partial q_1}, \dots, \frac{\partial}{\partial q_n} \ ; \ \frac{\partial}{\partial p_1}, \dots, \frac{\partial}{\partial p_n} \right) . \quad (4.20)$$

El sistema Hamiltoniano cuenta con n grados de libertad (movimiento), donde $\mathcal{J}$ es la matriz de Poisson de tamaño $2n \times 2n$ en (4.19),

$$\mathcal{J} = \begin{bmatrix} 0 & \mathcal{I} \\ -\mathcal{I} & 0 \end{bmatrix} , \quad (4.21)$$

con $\mathcal{I}$ la matriz identidad de tamaño $n \times n$. Las $2n$ ecuaciones del sistema dinámico que resultan de (4.19) se dividen en dos conjuntos con n ecuaciones cada uno para las variables en $\vec{r} = (q_i, p_i)$,

$$\dot{q}_i = \frac{\partial \mathcal{H}}{\partial p_i}; \quad \dot{p}_i = -\frac{\partial \mathcal{H}}{\partial q_i} , \quad (4.22)$$

$$i = 1, 2, \dots, n \quad (4.23)$$

donde q_i representa los grados de libertad y p_i los respectivos momentos canónicos conjugados; este conjunto de $2n$ —ecuaciones son conocidas como ecuaciones de Hamilton. Como ejemplo, para un sistema Hamiltoniano que solo posee un grado de libertad ($n = 1$), solo se genera un par de ecuaciones,

$$\dot{q}_1 = \frac{\partial \mathcal{H}}{\partial p_1}; \quad \dot{p}_1 = -\frac{\partial \mathcal{H}}{\partial q_1}; \quad (4.24)$$

mientras que para el caso de dos grados de libertad ($n = 2$) el conjunto de ecuaciones (4.19) es

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{p}_1 \\ \dot{p}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \partial_{q_1} \mathcal{H} \\ \partial_{q_2} \mathcal{H} \\ \partial_{p_1} \mathcal{H} \\ \partial_{p_2} \mathcal{H} \end{bmatrix}, \quad (4.25)$$

y por lo tanto,

$$\dot{q}_1 = \frac{\partial \mathcal{H}}{\partial p_1}; \quad \dot{p}_1 = -\frac{\partial \mathcal{H}}{\partial q_1}, \quad (4.26)$$

$$\dot{q}_2 = \frac{\partial \mathcal{H}}{\partial p_2}; \quad \dot{p}_2 = -\frac{\partial \mathcal{H}}{\partial q_2}. \quad (4.27)$$

En un sistema Hamiltoniano pueden existir cantidades $I(q, p)$ que se preservan durante la evolución temporal, estas son definidas como *integrales de movimiento* (constantes de movimiento). Cuando se tiene la misma cantidad de k integrales y n grados de libertad, ($n = k$), el sistema se denomina *integrable*, donde las constantes de movimiento están en involución (paréntesis de Poisson entre las cantidades igual a cero) y son algebraicamente independientes. Considerando entonces que $I(q, p)$ es integral de movimiento, se cumple que su derivada temporal es nula al igual que el paréntesis de Poisson de $\mathcal{I}$ con $\mathcal{H}$;

$$\frac{dI(q, p)}{dt} = \{I(q, p), \mathcal{H}\} = 0. \quad (4.28)$$

Cuando $k \neq n$, se tienen los casos siguientes; para $n < k < 2n - 1$, el sistema es

minimalmente superintegrable, por otro lado para $k = 2n - 1$ integrales, se tiene el caso de ser maximalmente superintegrable. Si $0 \leq k < n$ se denomina sistema no-integrable.

4.3.1. Sección de Poincaré

Para sistemas Hamiltonianos con n grados de libertad la representación en el espacio fase es $2n$ -dimensional con las variables canónicas q_i, p_i . Considerando el caso de dos grados de libertad ($n = 2$) el espacio fase de cuatro dimensiones se compone de las variables (q_1, p_1, q_2, p_2) . Si el sistema es conservativo, para cada condición inicial dada se conserva la energía total del sistema ($E \equiv \mathcal{H}$) y el espacio fase correspondiente es tridimensional donde uno de los momentos canónicos depende de la energía y de las otras variables dinámicas. Una forma de estudiar las orbitas en el espacio fase es considerar el corte de un plano en la superficie de energía constante. Este plano se conoce como *Sección de Poincaré* [5], el cual puede ser obtenido, considerando por ejemplo, $q_2 = 0$ y las cantidades q_1, p_1 sirven como coordenadas en este plano. Siguiendo una órbita (integrando numéricamente las ecuaciones de Hamilton), cada que esta pasa por el plano definido por $q_2 = 0$ se marcan las correspondientes coordenadas (q_1, p_1) por las que atraviesa el plano, formando así un mapeo de puntos sucesivos (Fig. 4.2). Cuando el sistema tiene movimientos ligados acotados, los puntos marcados en la superficie están contenidos en una cierta región del plano determinada por la energía.

Tomando diversas condiciones iniciales $X_0 = (x(t_0), p_x(t_0), y(t_0) = 0, p_y(t_0))$ en la superficie con cierta energía E , las intersecciones de la órbita proporcionan una especie de mapeo del movimiento en el espacio fase donde se pueden localizar los puntos fijos mencionados en la Sección 4.2.1 y se puede hacer un estudio cualitativo del sistema sin la solución analítica.

4.3.2. Líneas de Simetría

A continuación, se ofrece una descripción general de las líneas de simetría, formadas por mapeos discretos de puntos estacionarios, cuya intersección ayuda a determinar la ubicación de puntos en la superficie correspondientes a órbitas periódicas que también representan puntos fijos del sistema. En este trabajo se utiliza el enfoque adoptado por Piña-Jiménez [6] y usado en el artículo propio, [7], en el que se establece que las

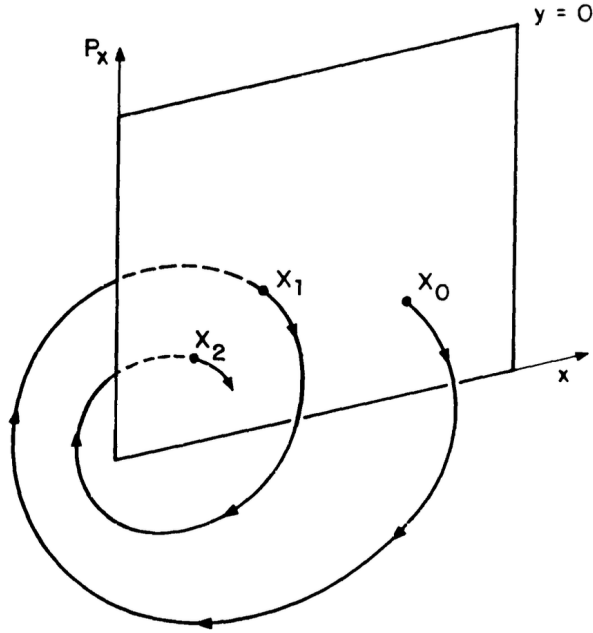


Figura 4.2: Sección de superficie determinada en este caso por $q_2 = y = 0$, donde los primeros cruces de la órbita en el plano $q_1 - p_1$ (en esta figura el plano es $x - p_x$) se ilustran como X_0, X_1, X_2 , donde $X_j = (x(t_j), p_x(t_j), y(t_j) = 0, p_y(t_j))$, $j = 0, 1, 2, \dots$.

líneas de simetría son parte de una familia de involuciones. La sección de Poincaré T , se puede expresar como la composición de dos involuciones de la forma:

$$T = I_0 T^{-1} I_0 = I_1 I_0, \text{ con } I_0^2 = I_1^2 = 1. \quad (4.29)$$

donde $T, I_0, I_1 : \Sigma \rightarrow \Sigma$, con Σ que denota la sección, ejemplos físicos de las involuciones I_0 y I_1 son las transformaciones de velocidad y de inversión temporal, respectivamente. Esta idea fue utilizada por Birkhoff para estudiar geodésicas y el problema de los tres cuerpos; también por De Vogelaere [8] para examinar el problema de Störmer de una carga en movimiento en un campo dipolar. Incorporando $I_j = T^j I_0$, el conjunto de transformaciones $T^j, I_i : \Sigma \rightarrow \Sigma$, con $i, j \in \mathbb{Z}$, forman un grupo infinito

de transformaciones discretas:

$$T^j I_k = I_{j+k} , \quad (4.30)$$

$$I_j I_k = T^{j-k} , \quad (4.31)$$

$$I_j T^k = I_{j-k} . \quad (4.32)$$

Las líneas de simetría Γ_i se definen como puntos estacionarios de I_i :

$$\Gamma_i : \{ \mathbf{x} \in S \mid I_i \mathbf{x} = \mathbf{x} \} , \text{ para toda } i \in \mathbb{Z} ; \quad (4.33)$$

de las propiedades (4.30)-(4.32) también se puede demostrar que las líneas de simetría se transforman con T^i ó I_i en otras líneas de simetría,

$$T^i \Gamma_j = \Gamma_{2i+j} , \quad (4.34)$$

$$I_i \Gamma_j = \Gamma_{2i-j} . \quad (4.35)$$

De particular importancia son las líneas fundamentales Γ_0 y Γ_1 que llevan a la propiedad:

$$T^i \Gamma_j = \Gamma_{2i+j} , \quad (4.36)$$

lo que implica que Γ_0 genera todas las líneas con índices pares $\Gamma_{2i} = T^i \Gamma_0$, mientras que Γ_1 , todas las impares $\Gamma_{2i+1} = T^i \Gamma_1$. Mediante (4.31) se puede demostrar que la intersección de las líneas de simetría, $\Gamma_i \cap \Gamma_j$, corresponden a órbitas periódicas, con el período siendo divisor del entero $|j - k|$. En la Fig. 4.3 se ilustran las líneas de simetría del mapa estándar, analizado en [6], el cual sirve como un modelo sencillo del mapa de Poincaré de un sistema Hamiltoniano con dos grados de libertad.

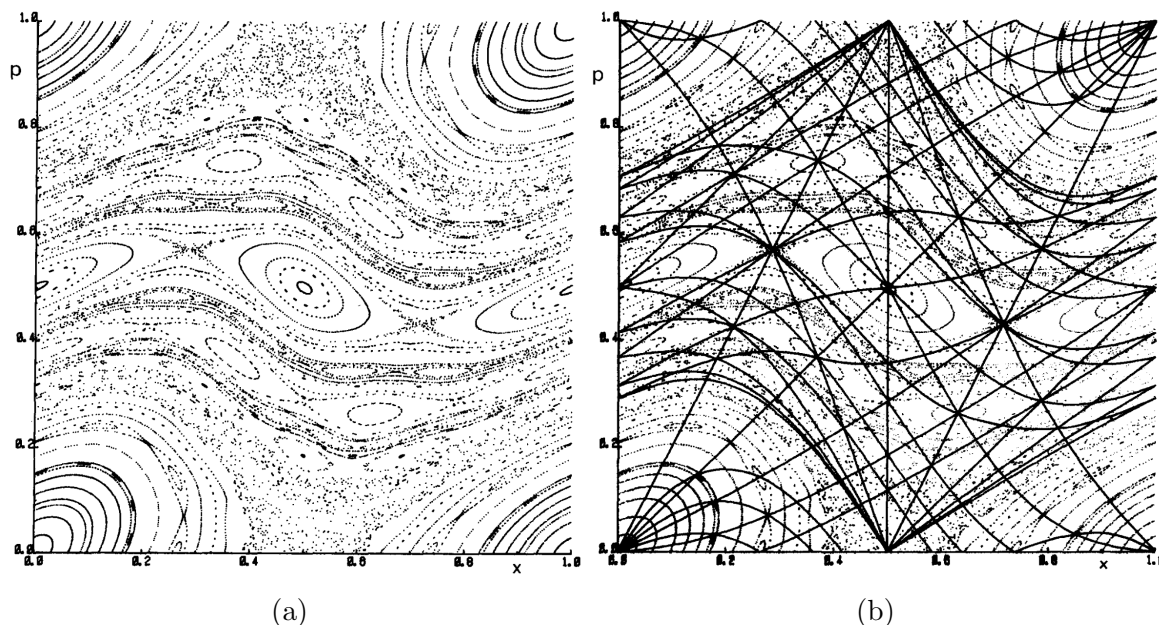


Figura 4.3: La representación de la sección de superficie de Poincaré para el mapa estándar en (a) exhibido en [6] y la superposición con algunas líneas de simetría (b) donde se puede apreciar que sus intersecciones denotan los puntos por donde pasan órbitas periódicas los cuales corresponden a puntos fijos centrales, Fig. 4.1 (g) y punto silla, Fig. 4.1 (h).

4.4. Métodos de integración numérica

Cuando no se puede calcular la solución analítica de las ecuaciones diferenciales debido a la complejidad que puede presentar $\vec{f}(\vec{x})$ en (4.4), los métodos numéricos son de gran ayuda para encontrar soluciones numéricas aproximadas. Los métodos Runge-Kutta, inicialmente desarrollados por Carl Runge y Wilhelm Kutta alrededor de 1900 [9], son una familia de algoritmos [10], de iteración a paso fijo para aproximar soluciones numéricas de una ecuación diferencial, así como el método de Euler, propuesto por Leonhard Euler en 1768 [11], también conocido como método Runge-Kutta de orden uno.

Existen extensiones a estos métodos, como el conocido método multipaso de Runge-Kutta-Fehlberg (RK-45) que tiene una mejor precisión [12] de la solución numérica, el cual se utiliza más adelante en este trabajo. En esta sección se describen solo algunos métodos, a manera de ilustración de como se construyen soluciones aproximadas me-

diante integración numérica con paso fijo, los cuales son de gran utilidad para construir por ejemplo las secciones de superficie de Poincaré (Sección.4.3.1) y líneas de simetría (Sección.4.3.2).

4.4.1. Método de Euler

Este es un método para resolver numéricamente una ecuación diferencial

$$\frac{dx}{dt} = f(x(t)) , \quad (4.37)$$

en el cual se considera que cada punto de la solución aproximada se obtiene de evaluar $f(x, t)$ para N valores discretizados de t en el intervalo $[t_0, t_f]$, con una separación Δt entre cada uno de ellos, como sigue

$$x_i = x_{i-1} + \Delta t f(x_{i-1}, t_{i-1}) , \quad (4.38)$$

donde

$$t_i = t_0 + i\Delta t \quad , \quad i = 0, 1, 2, \dots, N-1 , \quad (4.39)$$

y la separación entre cada elemento dado por

$$\Delta t = \frac{x_f - x_0}{N} , \quad (4.40)$$

y se define como el paso de integración de la solución numérica. La ecuación (4.38) se obtiene considerando la definición de la derivada

$$\frac{dx(t_k)}{dt} = \lim_{h \rightarrow \infty} \frac{x(t_k + \Delta t) - x(t_k)}{\Delta t} , \quad (4.41)$$

con un valor infinitesimal del paso de integración dt ,

$$\frac{dx(t_k)}{dt} = f(x(t_k)) \simeq \frac{x(t_k + \Delta t) - x(t_k)}{\Delta t} , \quad (4.42)$$

por lo tanto,

$$x(t_{k+1}) \simeq x(t_k) + \Delta t f(x(t_k)) \quad ; \quad k = 0, 1, \dots, N-1 , \quad (4.43)$$

esto último sugiere el cálculo de los puntos x_k mediante recurrencia partiendo de una condición inicial $x(t = 0) = x_0$. La continuidad se obtiene cuando $\Delta t \rightarrow \infty$. A manera de ilustración, continuación se integran numéricamente un par de ecuaciones diferenciales sencillas (de las cuales se puede calcular la solución analítica) usando el método de Euler desarrollado en un código de Python (presentados en el Apéndice A). Se comparan las soluciones numéricas con la soluciones exactas.

El primer ejemplo considera la solución numérica de la ecuación diferencial

$$x' = 0,1\sqrt{t} + 0,4t^2 , \quad (4.44)$$

usando como condición inicial $x(t = 0) = 0$ en el intervalo $[x_0 = 0, x_f = 50]$ con $N = 100$ y $N = 5000$, que corresponden al paso de integración $\Delta t = 0,5$ y $\Delta t = 0,01$, respectivamente. Se ilustra la solución aproximada y exacta para este ejemplo en la Fig.4.4. Otro ejemplo, para observar el efecto en la precisión de la integración al realizar un incremento en el número de pasos de integración, es la ecuación diferencial

$$x' = t^2 \sin(t) , \quad (4.45)$$

dentro del intervalo $[-\frac{\pi}{4}, 25]$, con la condición inicial $x(t = 0) = 0$ y el número de elementos $N = 30$, $N = 2000$ con pasos de integración $\Delta t = 0,85$ y $\Delta t = 0,012$, respectivamente. Una comparación de la solución aproximada y analítica se muestra en la Fig. 4.5.

En las Figuras 4.5, se hace notar una falta de precisión en la integración numérica para llegar a obtener los valores de la solución analítica; por lo tanto deben utilizarse otros métodos, como el que se describe a continuación, para encontrar una mejor aproximación.

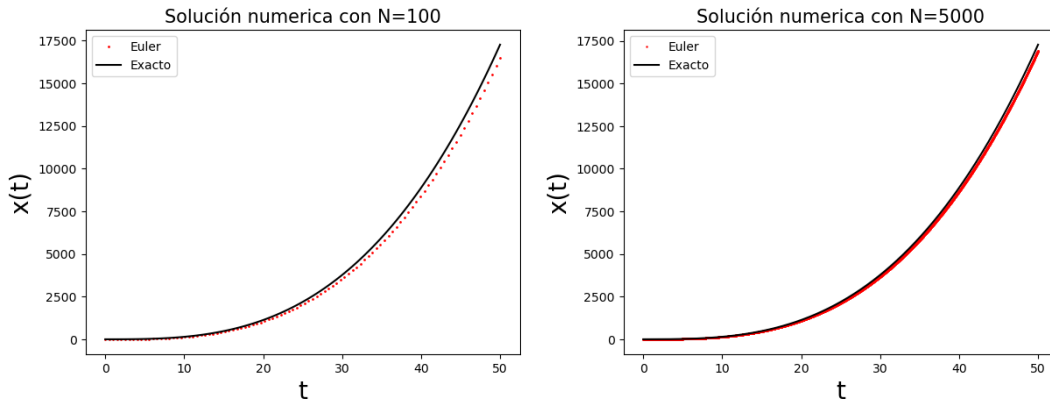


Figura 4.4: La solución numérica con el método de Euler de la ecuación (4.44) (línea roja punteada) se aproxima más a la solución exacta (línea negra solida) incrementando el valor de $N = 100$ a $N = 5000$ con pasos de integración $\Delta t = 0,5$ y $\Delta t = 0,01$, respectivamente.

4.4.2. Método de Runge-Kutta de orden 2 y orden 4

Para métodos Runge-Kutta, por ejemplo de segundo orden, se utilizan las pendientes en dos puntos diferentes para estimar el valor de la función en el siguiente paso de tiempo. Este método implica calcular dos pendientes, una al principio del intervalo y otra en el punto medio del intervalo, y luego combinarlas ponderadamente para obtener una mejor aproximación al valor de la función en el siguiente paso.; lo que realizaron Runge y Kutta [10] fue escribir el método de segundo orden con $\Delta t = t_{i+1} - t_i$ de forma

$$x_{i+1} = x_i + \Delta t(a_1 k_1 + a_2 k_2) , \quad (4.46)$$

donde se tiene que $k_1 = f(x_i, t_i) , \quad (4.47)$

$$k_2 = f(x_i + \Delta t k_1, t_i + \Delta t) . \quad (4.48)$$

Estas expresiones se obtienen de una expansión en serie de Taylor a segundo orden para y_n ,

$$y_{i+1} = y_i + \frac{dy}{dx}|_{x_i, y_i} h + \frac{1}{2!} \frac{d^2 y}{dx^2}|_{x_i, y_i} h^2 + O(h^3) ; \quad (4.49)$$

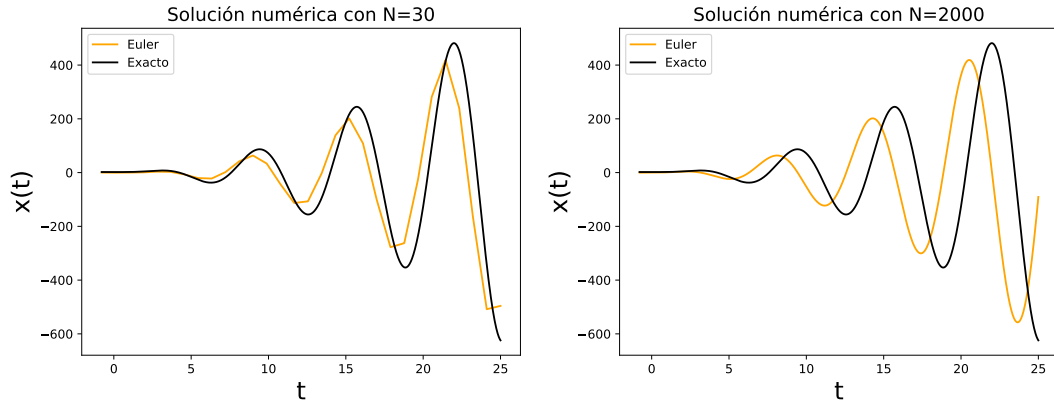


Figura 4.5: La solución numérica de la ecuación (4.45), un ejemplo ligeramente más complejo que el anterior, a pesar de incrementar el número de elementos en la integración numérica de $N = 30$ a $N = 2000$, no se aproxima a la solución exacta.

considerando la ecuación (4.37) se reescribe la serie de Taylor de una forma más compacta,

$$x_{i+1} = x_i + f(x_i, t_i)h + \frac{1}{2!}f'(x_i, t_i)\Delta t^2 + O(\Delta t^3) . \quad (4.50)$$

Dado que $f(x, y)$ depende de dos variables se considera la siguiente derivada de esta función

$$f'(x, t) = \frac{\partial f(x, t)}{\partial x} + \frac{\partial f(x, t)}{\partial t} \frac{dt}{dx} ; \quad (4.51)$$

por lo que la serie nos queda de nuevo como

$$x_{i+1} = x_i + f(x_i, t_i)h + \frac{1}{2!}\left(\frac{\partial f(x, t)}{\partial x}\bigg|_{x_i, t_i} + \frac{\partial f(x, t)}{\partial y}\bigg|_{x_i, t_i} \frac{dt}{dx}\right)\Delta t^2 + O(\Delta t^3) \quad (4.52)$$

$$= x_i + f(x_i, t_i)h + \frac{1}{2}\frac{\partial f(x, t)}{\partial x}\bigg|_{x_i, t_i}\Delta t^2 + \frac{\partial f(x, t)}{\partial t}\bigg|_{x_i, t_i}f(x_i, t_i)\Delta t^2 + O(\Delta t^3) . \quad (4.53)$$

Ahora los términos utilizados en el método de segundo orden de Runge-Kutta para k_2 pueden ser escritos como una expansión en serie de dos variables, los primeros tres términos de esta serie son

$$k_2 = f(x_i + \alpha\Delta t, t_i + \beta k_1\Delta t) \quad (4.54)$$

$$= f(x_i, t_i) + \alpha\Delta t \frac{\partial f(x, t)}{\partial x}\bigg|_{x_i, t_i} + \beta k_1\Delta t \frac{\partial f(x, t)}{\partial x}\bigg|_{x_i, t_i} + O(\Delta t^2) . \quad (4.55)$$

Por lo tanto

$$x_{i+1} = x_i + (a_1 k_1 + a_2 k_2) \Delta t \quad (4.56)$$

$$= x_i + \left[a_1 f(x_i, t_i) + a_2 \left(f(x_i, t_i) + \alpha \Delta t \frac{\partial f(x, t)}{\partial x} \Big|_{x_i, t_i} + \beta k_1 \Delta t \frac{\partial f(x, t)}{\partial x} \Big|_{x_i, t_i} + O(\Delta t^2) \right) \right] \Delta t \quad (4.57)$$

$$= x_i + (a_1 + a_2) \Delta t f(x_i, t_i) + a_2 \alpha \Delta t^2 \frac{\partial f}{\partial x} \Big|_{x_i, t_i} + a_2 \beta f(x_i, t_i) h^2 \frac{\partial f}{\partial t} \Big|_{x_i, t_i} + O(\Delta t^3) \quad (4.58)$$

Igualamos los términos de las ecuaciones (4.53) y (4.58) y se obtiene el siguiente sistema de ecuaciones

$$a_1 + a_2 = 1 \quad (4.59)$$

$$a_2 \alpha = \frac{1}{2} \quad (4.60)$$

$$a_2 \beta = \frac{1}{2} \quad (4.61)$$

Dado que tenemos tres ecuaciones y cuatro variables, fijamos el valor de una de las variables, las otras tres serán determinadas por las tres ecuaciones de arriba. Generalmente el valor de a_2 se escoge para evaluar las otras tres constantes, los valores típicos para a_2 son $1/2$, 1 y $2/3$, los cuales dan pie al método de Heun, el método del punto medio y el método de Ralston, respectivamente [10]. Tenemos muchas combinaciones para los valores de a_1 , a_2 , α y β ; para este caso se considera $a_1 = a_2 = 1/2$ junto con $\alpha = \beta = 1$. Con esta elección se tiene el método Runge-Kutta a segundo orden (RK-2), donde las ecuaciones (4.46)-(4.48) quedan de la siguiente forma

$$x_{i+1} = x_i + \Delta t \frac{(k_1 + k_2)}{2} , \quad (4.62)$$

$$k_1 = f(x_i, t_i) , \quad (4.63)$$

$$k_2 = f(x_i + \Delta t, x_i + k_1 \Delta t) . \quad (4.64)$$

Realizando el mismo procedimiento a cuarto orden, se hace la expansión en serie de Taylor, obteniendo x_{n+1} que depende de k_1, k_2, k_3 y k_4 :

$$x_{n+1} = x_n + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4) , \quad (4.65)$$

$$\begin{aligned}
k_1 &= f(x_n, t_n) , \\
k_2 &= f\left(x_n + \frac{\Delta t k_1}{2}, t_n + \frac{\Delta t}{2}\right) , \\
k_3 &= f\left(x_n + \frac{\Delta t k_2}{2}, t_n + \frac{\Delta t}{2}\right) , \\
k_4 &= f(x_n + \Delta t k_3, t_n + \Delta t) ,
\end{aligned} \tag{4.66}$$

donde el tiempo tiene incrementos,

$$t_{n+1} = t_n + \Delta t \tag{4.67}$$

con k_1 como la pendiente al inicio del intervalo usando solo y_n , (que sería el método de Euler, solo usando k_1). Y k_2 es la pendiente a la mitad del intervalo pero usando y y k_2 ; para k_3 tenemos que es de nuevo la pendiente a mitad de nuestro intervalo pero ahora usando y_n y k_2 . Por último, k_4 es la pendiente al final del intervalo usando x_n y k_3 .

Para ilustrar los métodos descritos en esta Sección, se resuelve numéricamente en Python la ecuación (4.45) y se presenta una comparación en la Fig. 4.6. El código para integrar numéricamente con estos métodos se encuentra en el Apéndice A.

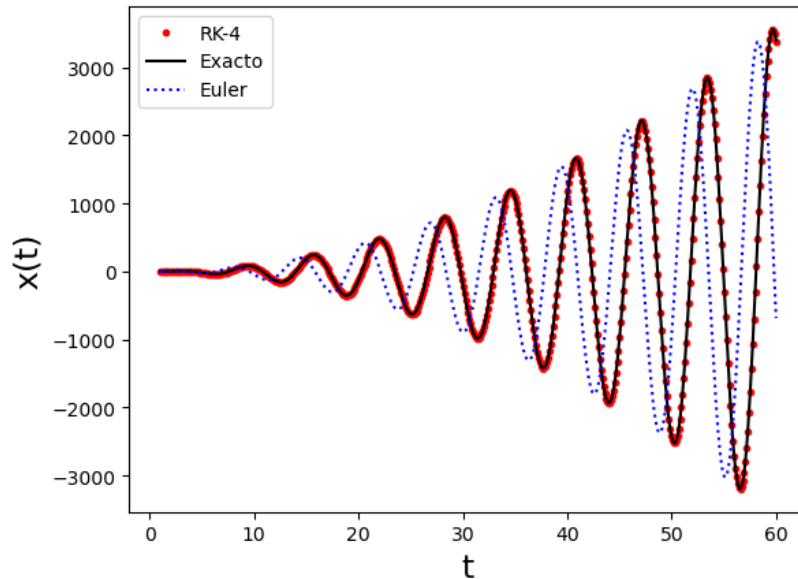


Figura 4.6: Comparación entre los métodos numéricos de Euler, Runge-Kutta de cuarto orden (RK-4) al integrar numéricamente la ecuación (4.45), con el paso de integración $dt = 0,117$.

En aprendizaje automático el problema de regresión se centra en identificar una función matemática (función objetivo) que relacione las variables de un conjunto de datos recopilados empíricamente o generados mediante simulaciones. Esta relación se obtiene mediante funciones de activación que son de forma lineal, polinomial, logarítmica, etc. Se debe modelar de manera precisa el comportamiento del sistema durante su evolución temporal, tanto para los datos con los que se construyó y también brindar información a tiempos posteriores (predicciones). Para evitar caer en un sobreajuste del modelo obtenido (lo cual sucede cuando el modelo no se generaliza correctamente a datos nuevos) o en un bajo ajuste (modelos simples que no reproducen la dinámica), se hace uso de métodos que determinen los pesos óptimos de la función objetivo. En este Capítulo se describe la regresión lineal con el método de mínimos cuadrados para encontrar un modelo de ecuaciones con los pesos β_i , $i = 1, 2, 3, \dots$

En términos generales, los algoritmos de aprendizaje automático (machine learning) giran en torno a métodos de regresión, los cuales a su vez necesitan de procesos como la regularización, para identificar las características relevantes del modelo, y optimización para tener al final modelos que sean parsimonios (precisos y sencillos) [13] ; algunos descritos en este Capítulo.

5.1. Regresión Lineal

El propósito principal de este proceso es realizar predicciones confiables sobre una cantidad de interés a partir de las variables independientes disponibles. Por ejemplo,

uno podría buscar determinar el precio de una propiedad residencial en función de variables como el año de construcción, el tamaño del terreno, entre otros factores. En este caso, el interés reside en encontrar el modelo óptimo que relacione de forma lineal las variables independientes x_i con la variable dependiente y de la forma

$$y = \beta_0 + \sum_{i=1}^n \beta_i x_i , \quad (5.1)$$

denominada como función objetivo y los pesos β_i . Una de las técnicas para resolver el problema es el caso de mínimos cuadrados, donde se buscan los pesos tal que (5.1) se ajuste lo mejor posible a los datos recolectados. El problema de regresión puede ser abordado con diferentes técnicas, como los mínimos cuadrados, mostrado a continuación, o por ejemplo usando el descenso de gradiente.

5.2. Mínimos Cuadrados

5.2.1. Caso simple

Este método busca encontrar los parámetros de la función objetivo (5.1) a manera que la suma de los cuadrados de las diferencias entre los valores observados y los valores predichos por el modelo sea mínima. En esencia, busca minimizar el error cuadrático medio (MSE) entre los datos reales y los estimados por la función lineal. Este enfoque nos da el valor de los pesos que mejor ajustan los datos al modelo al minimizar una medida de error específica. De esta manera, los mínimos cuadrados proporcionan una forma sistemática y eficiente para realizar regresión lineal relacionando variables dependientes e independientes.

Entonces, tomando el caso de una sola variable independiente en (5.1)

$$y = \beta_0 + \beta_1 x , \quad (5.2)$$

para encontrar los parámetros óptimos β_0 y β_1 , se considera minimizar el error cuadrático medio definido como

$$\epsilon_{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 , \quad (5.3)$$

que es el promedio de las diferencias cuadráticas entre los datos reales y_i y los datos

estimados $\hat{y}_i$ calculados con la función objetivo (5.2). Reescribiendo el error cuadrático con (5.2) se tiene la expresión

$$\epsilon_{MSE} = \sum_{i=1}^n (y_i - (\beta_1 x_i + \beta_0))^2 , \quad (5.4)$$

la cual se minimiza en función de los pesos β_0, β_1 mediante

$$\frac{\partial \epsilon_{MSE}}{\partial \beta_1} = 0 , \quad (5.5)$$

$$\frac{\partial \epsilon_{MSE}}{\partial \beta_0} = 0 . \quad (5.6)$$

Así que usando (5.4) en estas expresiones de minimización se obtienen,

$$\frac{\partial \epsilon_{MSE}}{\partial \beta_1} = -2 \sum_{i=1}^n (y_i - (\beta_1 x_i + \beta_0)) x_i = 0 , \quad (5.7)$$

$$\frac{\partial \epsilon_{MSE}}{\partial \beta_0} = -2 \sum_{i=1}^n (y_i - (\beta_1 x_i + \beta_0)) = 0 , \quad (5.8)$$

ordenando estas ecuaciones,

$$\begin{aligned} \sum_{i=1}^n y_i x_i &= \sum_{i=1}^n \beta_1 x_i^2 + \beta_0 \sum_{i=1}^n x_i , \\ \sum_{i=1}^n y_i &= \sum_{i=1}^n \beta_1 x_i + \beta_0 n . \end{aligned} \quad (5.9)$$

Dividiendo entre el número total de mediciones n , la segunda ecuación en (5.9), llegamos a

$$\bar{Y} = \beta_1 \bar{X} + \beta_0 , \quad (5.10)$$

donde se han introducido los promedios $\bar{Y}$ y $\bar{X}$:

$$\bar{Y} = \frac{\sum_{i=1}^n y_i}{n} , \quad (5.11)$$

$$\bar{X} = \frac{\sum_{i=1}^n x_i}{n} . \quad (5.12)$$

Despejando β_0 de (5.10):

$$\beta_0 = \bar{Y} - \beta_1 \bar{X}, \quad (5.13)$$

sustituimos (5.13) en la primera ecuación de (5.9),

$$\sum_{i=1}^n y_i x_i = \sum_{i=1}^n \beta_1 x_i^2 + (\bar{Y} - \beta_1 \bar{X}) \sum_{i=1}^n x_i \quad (5.14)$$

$$\sum_{i=1}^n (y_i x_i - \bar{Y} x_i) = \sum_{i=1}^n \beta_1 (x_i^2 - \bar{X} x_i) \quad (5.15)$$

$$\beta_1 = \frac{\sum_{i=1}^n (y_i x_i - \bar{Y} x_i)}{\sum_{i=1}^n (x_i^2 - \bar{X} x_i)} \quad (5.16)$$

$$\beta_1 = \frac{\sum_{i=1}^n x_i (y_i - \bar{Y})}{\sum_{i=1}^n x_i (x_i - \bar{X})}. \quad (5.17)$$

El valor de β_1 en (5.16), se puede escribir también de la siguiente forma; primero para el numerador se tiene,

$$\begin{aligned} & \sum y_i x_i - \bar{Y} \sum x_i \\ &= \sum y_i x_i - \bar{Y} \sum x_i - \sum \bar{X} \bar{Y} + \sum \bar{Y} \bar{X} \\ &= \sum y_i x_i - \bar{Y} \sum x_i - \bar{X} \sum y_i + \sum \bar{Y} \bar{X} \\ &= \sum (y_i x_i - y_i \bar{X} - \bar{Y} x_i + \bar{Y} \bar{X}) \\ &= \sum (y_i - \bar{Y})(x_i - \bar{X}); \end{aligned} \quad (5.18)$$

y para el denominador,

$$\begin{aligned} & \sum x_i^2 - \sum \bar{X} x_i \\ &= \sum x_i^2 - \sum \bar{X} x_i + \sum x_i \bar{X} - \sum x_i \bar{X} \\ &= \sum x_i^2 - 2 \sum \bar{X} x_i + \sum x_i \bar{X}^2 \\ &= \sum (x_i^2 - 2 \bar{X} x_i + x_i \bar{X}^2) \\ &= \sum (x_i - \bar{X})^2. \end{aligned} \quad (5.20)$$

$$(5.21)$$

Sustituyendo (5.19) y (5.21) en (5.16),

$$\beta_1 = \frac{\sum (y_i - \bar{Y})(x_i - \bar{X})}{\sum (x_i - \bar{X})^2}, \quad (5.22)$$

finalmente dividiendo entre el número total de mediciones n en el numerador y denominador, se obtiene la relación:

$$\beta_1 = \frac{\frac{\sum (y_i - \bar{Y})(x_i - \bar{X})}{n}}{\frac{\sum (x_i - \bar{X})^2}{n}} = \frac{S_{xy}}{S_x^2}, \quad (5.23)$$

donde S_{xy} representa la covarianza y S_x^2 la varianza.

5.2.2. Ejemplo en Python: regresión con una variable independiente

Se presenta un ejemplo ilustrativo, desarrollado en Python, de la regresión lineal por mínimos cuadrados para calcular la esperanza de vida, definido por la variable independiente y , de una persona a partir del índice de masa corporal (BMI), como la variable independiente x . Usando datos que se pueden descargar en línea¹, se obtiene la función objetivo $y = \beta_0 + \beta_1 x$, Fig. 5.1, usando las ecuaciones (5.13) y (5.23) para encontrar los pesos β_0 , β_1 , junto con el error cuadrático medio (ϵ_{MSE}) que presenta el modelo resultante.

5.2.3. Caso multivariable

Considerando más de una variable independiente en (5.1), ahora se cuenta con $x_1, x_2, \dots, x_k$ y los pesos $\beta_0, \beta_1, \dots, \beta_k$, por lo que la ecuación de ajuste es

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k; \quad (5.24)$$

Esta última ecuación es la forma compacta de la idea sobre una neurona artificial propuesto por McCullough y Walter Pitts, con función de activación lineal, la cual se muestra en la Figura 5.2. Cuando se ajusta un modelo de regresión y este cuenta con

¹<https://www.gapminder.org>

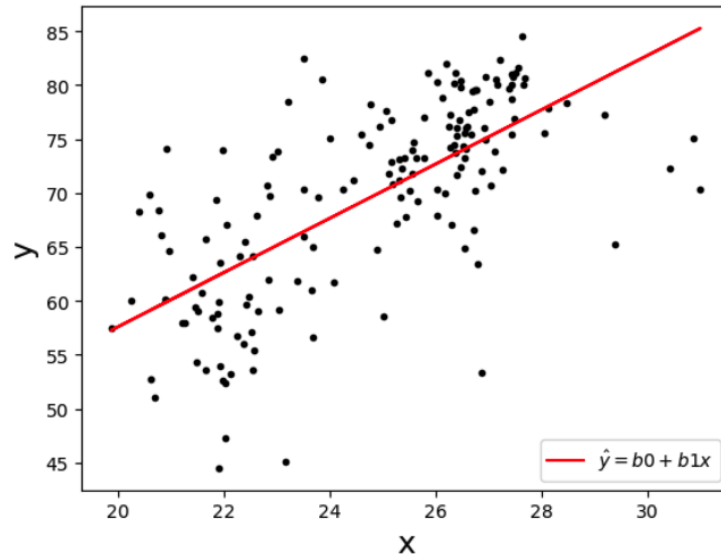


Figura 5.1: Representación de la recta de ajuste (función objetivo) obtenida por regresión lineal con los pesos $\beta_0 = 7,23$, $\beta_1 = 2,52$, para calcular la esperanza de vida, y , a partir del índice de masas corporal, x . Claramente este no es un buen modelo de ajuste pero ilustra las ideas presentadas sobre regresión lineal.

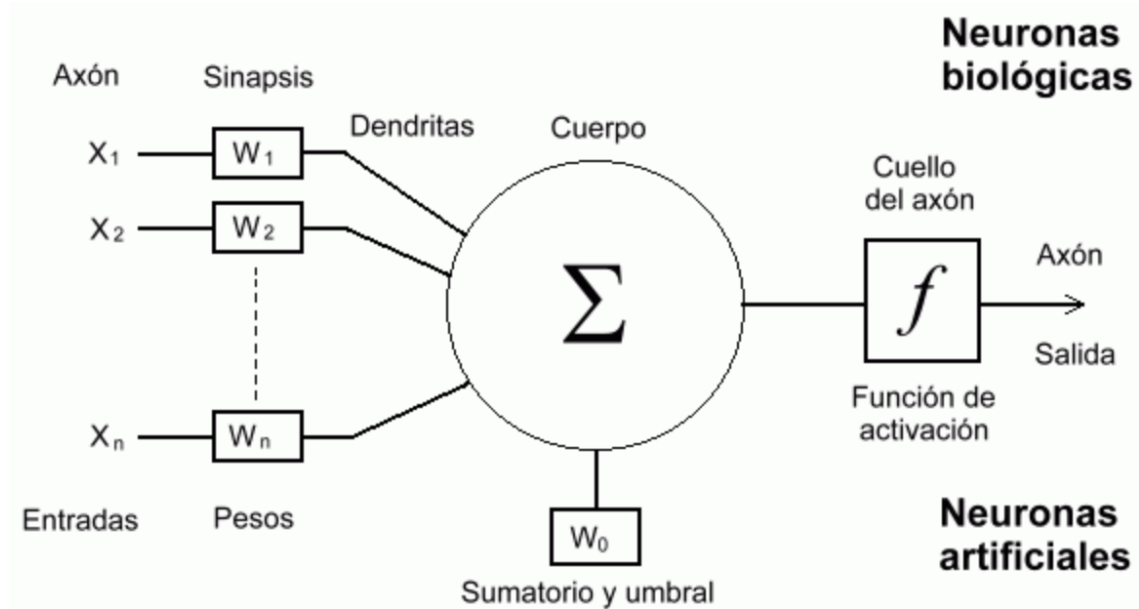


Figura 5.2: Representación gráfica de la estructura de una neurona artificial, los pesos de la ecuación (5.24) aquí se representan por w_n ; se muestra la comparación entre las partes de una neurona biológica y lo artificial. El uso de diversas neuronas se desarrolla en el aprendizaje profundo (deep learning).

más ecuaciones que variables independientes ($n > k$) el sistema se denomina sobre-determinado; en caso contrario, cuando $n < k$ se define como subdeterminado. Así como en la sección anterior, se busca minimizar la suma del error cuadrático (5.3):

$$\epsilon_{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (5.25)$$

para el cual, en el caso multivariable dicha suma debe considerar explícitamente que para cada dato estimado $\hat{y}$:

$$\begin{cases} \hat{y}_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \cdots + \beta_k x_{1k} \\ \vdots \\ \hat{y}_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} \\ \vdots \\ \hat{y}_n = \beta_0 + \beta_1 x_{n1} + \beta_2 x_{n2} + \cdots + \beta_k x_{nk} . \end{cases} \quad (5.26)$$

Por simplicidad, se puede escribir (5.26) de forma matricial como $\hat{\mathbf{Y}} = \mathbf{XB}$. Para resolver el problema de mínimos cuadrados, se escribe la diferencia entre los datos reales y los estimados de la forma,

$$\bar{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix} = \begin{pmatrix} y_1 - \hat{y}_1 \\ y_2 - \hat{y}_2 \\ \vdots \\ y_n - \hat{y}_n \end{pmatrix}, \quad (5.27)$$

y sustituyendo (5.26) en (5.27),

$$\bar{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix} = \begin{pmatrix} y_1 - (\beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \cdots + \beta_k x_{1k}) \\ y_2 - (\beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \cdots + \beta_k x_{2k}) \\ \vdots \\ y_n - (\beta_0 + \beta_1 x_{n1} + \beta_2 x_{n2} + \cdots + \beta_k x_{nk}) \end{pmatrix}, \quad (5.28)$$

se tiene la siguiente expresión matricial,

$$\bar{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_0 \\ \vdots \\ y_n \end{pmatrix} - \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & & & \ddots & \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}, \quad (5.29)$$

por lo tanto

$$\bar{\epsilon} = \mathbf{Y} - \mathbf{XB}. \quad (5.30)$$

Para encontrar los valores de la matriz $\mathbf{B}$ (pesos β_k en la función objetivo) que minimicen el error cuadrático $\bar{\epsilon}_{MSE}$,

$$\epsilon_{MSE} = \frac{1}{n} \sum_{i=1}^n \bar{\epsilon}^2, \quad (5.31)$$

se resuelve el problema de optimización siguiente

$$\min[\epsilon_{MSE}] = \min \left[\begin{pmatrix} \epsilon_1 & \epsilon_2 & \cdots & \epsilon_n \end{pmatrix} \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix} \right] = \min [\bar{\epsilon}^T \bar{\epsilon}]. \quad (5.32)$$

Para encontrar la matriz $\mathbf{B}$ en (5.30) que resuelve (5.32), se usa la siguiente condición de minimización,

$$\frac{\partial(\epsilon^T \epsilon)}{\partial \mathbf{B}} = 0, \quad (5.33)$$

y la solución del problema es

$$\mathbf{B} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}. \quad (5.34)$$

El calculo para obtener (5.34) se puede consultar en el Apéndice B. Al resolver problemas de regresión múltiple en python, existen dos opciones, realizar las operaciones entre matrices para encontrar la solución (5.34) o bien usando una librería de Python (Scikitlearn²) que contiene una opción para calcular modelos de regresión con solo

²<https://scikit-learn.org/stable/>

escribir algunas de líneas de código.

5.2.4. Coeficiente de correlación

Una vez realizado el método de mínimos cuadrados, se busca disponer de algún indicador sobre el grado de precisión con el que la recta $\hat{y}$ se ajusta a nuestros datos. Para esto tenemos el coeficiente de correlación, el cual se basa en la descomposición de la varianza de la variable y_i a la que se le llama *varianza total*. Para realizar esta descomposición consideramos el error entre cada punto real y_i y el obtenido con el modelo $\hat{y}_i$ como

$$y_i = \epsilon_i + \hat{y}_i , \quad (5.35)$$

restando de ambos lados $\bar{Y}$ se tiene:

$$y_i - \bar{Y} = \epsilon_i + \hat{y}_i - \bar{Y} , \quad (5.36)$$

elevando al cuadrado ambos lados de la igualdad

$$(y_i - \bar{Y})^2 = (\epsilon_i + (\hat{y}_i - \bar{Y}))^2 \quad (5.37)$$

$$= (\hat{y}_i - \bar{Y})^2 + \epsilon_i^2 - 2\epsilon_i(\hat{y}_i - \bar{Y}) . \quad (5.38)$$

Considerando la sumatoria en todos los pares de datos

$$\sum_{i=1}^n (y_i - \bar{Y})^2 = \sum_{i=1}^n (\hat{y}_i - \bar{Y})^2 + \sum_{i=1}^n \epsilon_i^2 - 2 \sum_{i=1}^n \epsilon_i (\hat{y}_i - \bar{Y}) , \quad (5.39)$$

donde el último término es idénticamente igual a cero

$$2 \sum_{i=1}^n \epsilon_i (\hat{y}_i - \bar{Y}) = 2 \sum_{i=1}^n \epsilon_i \hat{y}_i - 2 \sum_{i=1}^n \epsilon_i \bar{Y} \quad (5.40)$$

$$= 2 \sum_{i=1}^n \epsilon_i \bar{Y} - 2 \sum_{i=1}^n \epsilon_i \bar{Y} = 0 . \quad (5.41)$$

Entonces con este resultado para (5.39); dividiendo cada término por el número total de mediciones n y usando 5.3 ,

$$\frac{\sum_{i=1}^n (y_i - \bar{Y})^2}{n} = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{Y})^2}{n} + \frac{\sum_{i=1}^n \epsilon_i^2}{n} \quad (5.42)$$

$$\underbrace{\frac{\sum_{i=1}^n (y_i - \bar{Y})^2}{n}}_{\text{Varianza Total}} = \underbrace{\frac{\sum_{i=1}^n (\hat{y}_i - \bar{Y})^2}{n}}_{\text{Varianza de Ajuste}} + \underbrace{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}}_{\text{Varianza Residual}} . \quad (5.43)$$

Por lo tanto se define el coeficiente de correlación como

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{Y})^2}{\sum_{i=1}^n (y_i - \bar{Y})^2} \quad (5.44)$$

que se define como la proporción de la varianza total y la varianza que resulta del ajuste lineal.

5.2.5. Ejemplo caso multivariable (β_1, β_2)

Como ejemplo de aplicación de la teoría descrita anteriormente para el caso de varias variables independientes y el uso de la librería en Scikitlearn de Python, se utiliza una base de datos³, con la cual se puede construir un modelo para predecir el costo de una vivienda. En este ejemplo se consideran como variables independientes, la edad de la casa x_1 y el número de tiendas de conveniencia x_2 con la función objetivo $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$ de la cual se buscan los pesos óptimos. El código propio, desarrollado para calcular (5.34) en este ejemplo multivariable se puede encontrar en el Apéndice(B.3). El uso de la librería es trivial y su desarrollo puede encontrarse vía internet. Los resultados obtenidos mediante código propio y con Scikitlearn son los mismos, estos se muestran en el Cuadro 5.1.

³<https://github.com/subashgandyer/datasets/blob/main/Real%20estate.csv>

	Libreria Scikitlearn	Método Matricial
β_0	32.02251474	32.02251474
β_1	-0.28601262	-0.28601262
β_2	2.69251293	2.69251293
ϵ	113.90655270539162	113.90655270539162
R^2	0.383253162	0.383253162

Cuadro 5.1: Comparación de los resultados obtenidos de forma numérica y con el uso de librerías para un sistema de dos variables independientes.

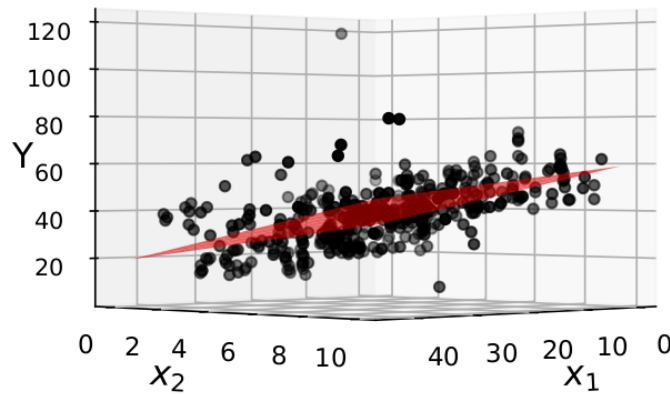


Figura 5.3: Hiperplano obtenido para la función $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$ mediante regresión lineal multivariable.

5.3. Regularización

En el contexto del ajuste de modelos lineales, la regularización es un enfoque para controlar y evitar el sobreajuste del modelo a los datos de entrenamiento, que se ocasiona cuando el modelo se ajusta excesivamente a los detalles específicos de los datos de entrenamiento (sobreajuste), lo que puede resultar en una generalización deficiente para nuevos datos.

La regularización se logra mediante la inclusión de términos adicionales a la función

de pérdida (error) que el modelo intenta minimizar durante el proceso de entrenamiento [14], [15]. Estos términos adicionales penalizan los pesos en la función objetivo, lo que ayuda a suavizar la función y a evitar que se ajuste excesivamente a los datos de entrenamiento. En el aprendizaje automático se debe hacer elección de los hiperparámetros del modelo, que son valores que se establecen antes de iniciar el proceso de entrenamiento del modelo y no son modificados después, estos controlan aspectos del proceso de entrenamiento y la complejidad del modelo, y pueden influir significativamente en el rendimiento y la capacidad de generalización del modelo. A diferencia de parámetros como los pesos del modelo, que se aprenden durante el entrenamiento a partir de los datos recolectados, los hiperparámetros deben especificarse externamente y generalmente se establecen a través de prueba y error, validación cruzada u otras técnicas de optimización, algunas de ellas se mencionan más adelante. A continuación, se presentan algunos de los métodos más comunes para aplicar regularización en modelos de regresión.

- **Least Absolute Shrinkage and Selection Operator (LASSO).** Se hace uso de la norma L_1 que consiste en agregar un término proporcional a la suma de los valores absolutos de los coeficientes de los pesos [15],

$$\epsilon_{LASSO} = \underbrace{\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2}_{\text{Error cuadrático (MSE)}} + \underbrace{\lambda \sum_{j=1}^p |\beta_j|}_{\text{norma } L_1}, \quad (5.45)$$

donde λ es un hiperparámetro de regularización que se puede seleccionar de forma aleatoria o bajo procesos de selección. Agregar la norma tiene el efecto de forzar algunos coeficientes a cero, lo que conduce a la selección de características y simplifica el modelo, obteniendo así un modelo de regresión dispersa.

- **Ridge Regression (RR).** Aquí el término es proporcional a la suma de los cuadrados de los coeficientes de los pesos [15],

$$\epsilon_{RR} = \underbrace{\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2}_{\text{Error cuadrático (MSE)}} + \underbrace{\lambda \sum_{j=1}^p \beta_j^2}_{\text{norma } L_2}; \quad (5.46)$$

como en el caso anterior λ es un hiperparámetro que se debe seleccionar antes de iniciar el ajuste del modelo. Aquí se reduce el impacto de algunas variables en el modelo por igual, aunque se puede simplificar el modelo, la diferencia con LASSO radica en que los pesos nunca se anulan.

- **Sequential Thresholded Least-Squares(STLQ).** En cualquiera de las formas de los datos recolectados en las series temporales, para resolver el problema de regresión se puede utilizar la técnica de regularización LASSO, descrito anteriormente, pero este tiene un alto costo computacional para conjuntos de datos de gran escala. En su lugar se hace uso del algoritmo denominado STLQ (sequential thresholded least-squares) [16], que implementa el método de mínimos cuadrados con el umbral secuencial λ que elimina algunos términos de la solución que sean menores a dicho valor. Este algoritmo secuencial da como resultado una representación dispersa, el procedimiento se resumen en lo siguiente:

- 1) Se calcula una solución con regresión por mínimos cuadrados para encontrar Ξ .
- 2) Para todos los pesos del primer calculo de regresión, se actualiza la matriz Ξ donde los pesos con valores menores que λ , se igualan a cero.
- 3) Una vez que se identifican los índices de los pesos no nulos restantes, obtenemos otra solución de mínimos cuadrados para Ξ sobre los índices restantes.
- 4) Los nuevos coeficientes se vuelven a someter al umbral λ , y el procedimiento continúa hasta que los coeficientes no nulos converjan.

El valor de λ puede establecerse de forma aleatoria o mediante un análisis de validación cruzada, dicha técnica se describe en la siguiente sección. En la Fig.5.4 se muestra el código de Matlab para este algoritmo mostrado en el anexo de [16]. Como parte del proceso de aprendizaje en programación, se construyó la correspondiente versión de este algoritmo en Python aplicado al ejemplo de un modelo lineal; este se presenta a continuación.

Ejemplo de STLQ

A manera de ilustración del algoritmo STLQ se realiza la construcción de un modelo lineal disperso usando una base de datos aleatoria en Python (código se

```

%% compute Sparse regression: sequential least squares
Xi = Theta\dXdt; % initial guess: Least-squares

% lambda is our sparsification knob.
for k=1:10
    smallinds = (abs(Xi)<lambda); % find small coefficients
    Xi(smallinds)=0; % and threshold
    for ind = 1:n % n is state dimension
        biginds = ~smallinds(:,ind);
        % Regress dynamics onto remaining terms to find sparse Xi
        Xi(biginds,ind) = Theta(:,biginds)\dXdt(:,ind);
    end
end
end

```

Figura 5.4: Algoritmo de la regularización STLQ descrito en el Anexo de [16] que promueve la regresión dispersa.

muestra en el Apéndice C) correspondiente a variables independientes definidas como x_1, x_2, x_3 para el calculo de una variable independiente y . El valor del umbral λ en este caso es aleatorio ya que solo se busca ilustrar la aplicación del algoritmo, aunque podría estimarse con métodos de optimización (descritos en la siguiente Sección). En este ejemplo el modelo lineal tiene la forma,

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 . \quad (5.47)$$

	Mínimos Cuadrados	Regresión Dispersa
β_0	121.70448779	121.70448779
β_1	2.94932354	2.94932354
β_2	0.27556962	0
β_3	-7.84335821	-7.84335821
ϵ_{MSE}	14922.84819	728.26575
R^2	-4.72922	0.95392

Cuadro 5.2: Comparación entre los pesos obtenidos por mínimos cuadrados y el algoritmo de regularización STLQ para el modelo lineal en (5.47). Con regresión dispersa se obtiene una precisión significativa en términos del coeficiente de correlación ($R \approx 1$) a comparación del uso de mínimos cuadrados donde se indica que el modelo no es funcional; de igual forma el valor del error cuadrático disminuye.

El algoritmo de la representación dispersa busca determinar cuáles parámetros

de ajuste β_k son iguales a cero para los cuales la variable relacionada a él no es significativa para el modelo que representa la dinámica del sistema. En este ejemplo, se observa que después de tres iteraciones el algoritmo converge, es decir, los valores de los pesos después de cierto número de iteraciones no presentan variación alguna, el parámetro β_2 se reduce a cero y por lo tanto la variable de estado correspondiente no influye en el modelo. Por ende las predicciones que se hagan con el modelo no dependen de la variable independiente x_2 .

5.3.1. Optimización

El proceso de optimización tiene como objetivo encontrar un modelo que sea *parsimonioso*[13], es decir que sea preciso y sencillo para lograr una generalización correcta y así poder conocer el comportamiento de un sistema en evolución. Con un modelo sencillo se hace referencia a que todas las características del sistema estén contenidas dentro de una función objetivo con pocos términos (pesos) y preciso en términos de que al realizar una interpolación o extrapolación de los datos, el error cuadrático medio obtenido sea el menor a comparación con otros modelos y el coeficiente de correlación se encuentre en el intervalo $R^2 \in [-1, 1]$.

Por otra parte obtener un modelo óptimo de regresión, implica ajustar de inicio los hiperparámetros, aquellos valores que se establecen antes de iniciar el entrenamiento en un algoritmo de aprendizaje automático; por ejemplo λ en el caso de LASSO que juega un papel importante para la penalización de los términos en el modelo o también el número de capas de una red neuronal, etc. Estos se pueden escoger a través de ensayo y error, lo cual tardaría bastante tiempo, por lo que es fundamental el uso de algunos métodos de *validación cruzada* como por ejemplo K-Fold, división de series de tiempo (Time series split) o el *frente de Pareto*, junto con las técnicas de regularización que controlen el sobreajuste y promuevan la generalización del modelo.

K-Fold

Esta técnica, referida como validación cruzada K-fold, se emplea en el ámbito del aprendizaje automático para la evaluación del rendimiento de un modelo o la selección de hiperparámetros con mayor precisión [17]. Consiste en la partición del conjunto de datos en K subconjuntos (o pliegues) de igual tamaño; por ejemplo, se puede considerar

dividir los datos en pliegues, cada uno comprendiendo el 10 % del total del conjunto. Manteniendo ciertos hiperparámetros fijos, se entrena un modelo utilizando $K - 1$ pliegues como conjunto de entrenamiento, y luego se valida este modelo utilizando el pliegue restante, calculando el error entre los datos de entrenamiento y los estimados. Este proceso se repite para cada uno de los K pliegues de datos, y finalmente se calcula el promedio de los errores obtenidos por cada modelo. Se selecciona aquel modelo, junto con sus respectivos hiperparámetros, que minimiza el error promedio. La validación cruzada K-fold (ilustrado en la Fig.5.5) contribuye a mitigar el sesgo en la selección de datos y proporciona una evaluación más sólida del rendimiento del modelo en datos no observados.

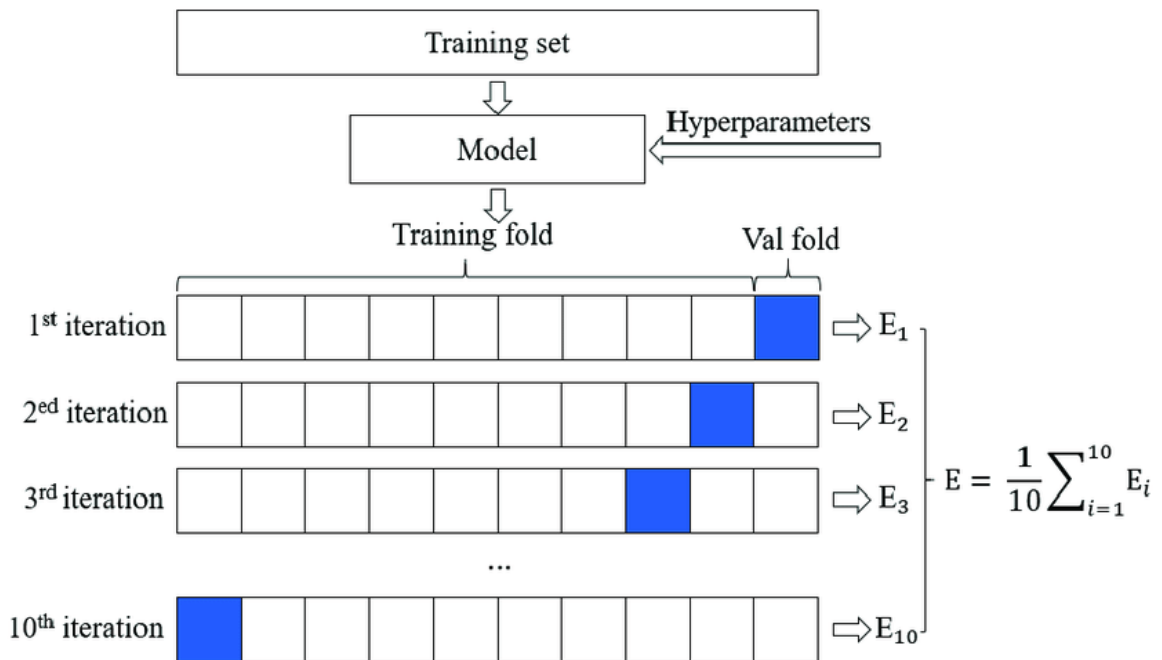


Figura 5.5: Ejemplo de la implementación para validación cruzada tipo K-fold con $K = 10$.

Frente de Pareto

Este es un concepto de la optimización multiobjetivo que describe soluciones no dominadas, estas son aquellas para las cuales no se puede mejorar un objetivo sin empeorar otro. En el contexto del ajuste de modelos por regresión utilizados para generar

un aprendizaje automático de una máquina, se establece que los modelos deben ser *parsimonios* [18], por lo que nuestros objetivos serían el menor error y número de términos en la función que describe al sistema. En la Figura 5.6 se ilustran para diferentes modelos su relación entre el número de términos que poseen y el error correspondiente, aunque muchos de ellos no cumplen con lo deseado (puntos verdes), existen otros que sí cumplen y se encuentran en la definida frontera de Pareto (puntos morados). Pero los modelos a considerar son aquellos que se encuentran en la región óptima de Pareto, región en donde no se cuenta con demasiado error ni un gran número de términos en el modelo. En el contexto de la regresión y la selección de características, LASSO (Least Absolute Shrinkage and Selection Operator) es una técnica que utiliza la regularización L_1 para promover la dispersión de la solución mediante la elección de un λ óptimo. Al agregar una penalización a la función de pérdida (error), LASSO tiende a forzar algunos coeficientes de los parámetros del modelo a cero, lo que resulta en la selección de un subconjunto de características importantes y, por lo tanto, en una solución dispersa capaz de interpolar el modelo obtenido a los datos de entrenamiento y extrapolar a datos nuevos de forma precisa. Por lo tanto el *frente de Pareto* considera el cálculo de modelos para diferentes valores en los hiperparámetros, como lo puede ser λ en el caso de usar LASSO o STLQ, con un 70 % de los datos como el conjunto de entrenamiento y el otro 30 % como el conjunto de validación/prueba (estos porcentajes pueden presentar variaciones), de los cuales se calcula el error entre las mediciones originales con las estimadas del modelo y se analiza el número de términos presentes en él. Finalmente se hace la elección del modelo que cumpla con la condición de ser parsimonioso. Una forma alternativa del frente de Pareto es realizar el mismo procedimiento descrito anteriormente, la única diferencia es que la elección del modelo óptimo es aquel que presenta el menor error en función del valor del hiperparámetro tanto para interpolación como extrapolación de datos, se ilustra esto último en la Figura 5.7 para el ejemplo de un sistema de osciladores cúbicos (ejemplo presentado en el siguiente Capítulo), usando escala logarítmica para el eje correspondiente al error del modelo.

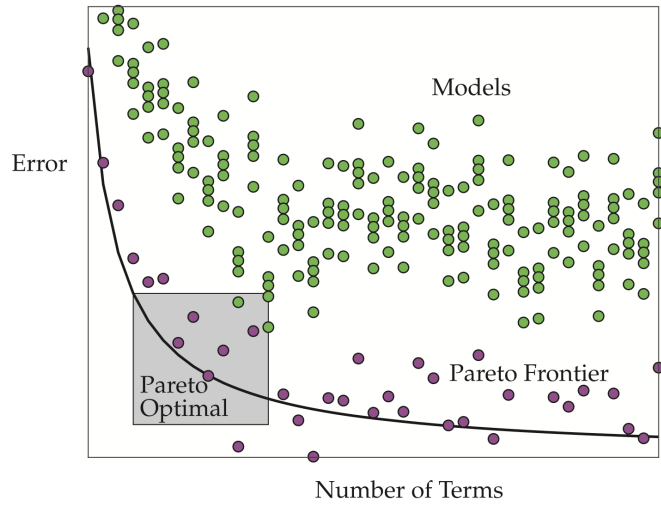


Figura 5.6: Elección de un modelo *parsimonio* mediante el frente de Pareto [18]. La región marcada en gris hace referencia a la zona óptima de Pareto, donde se encuentran los modelos con el menor error y número de términos los cuales capturan la esencia de la dinámica.

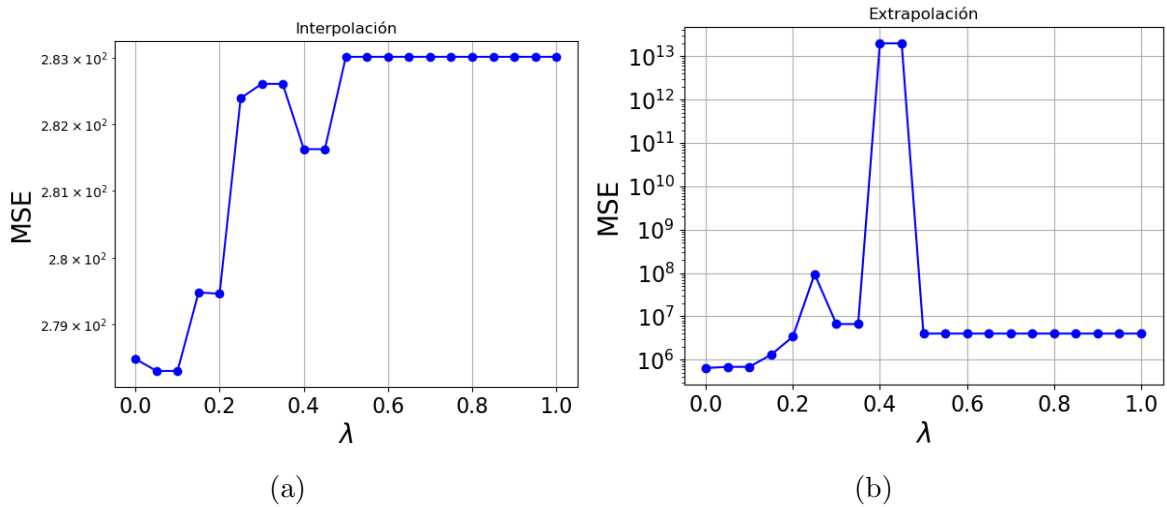


Figura 5.7: Gráficas en escala logarítmica de la interpolación (a) y extrapolación (b) de las series de tiempo, donde se compara el error cuadrático medio MSE (ϵ_{MSE}) de los modelos obtenidos con diferentes valores del hiperparámetro λ en la regularización con STLQ. Se selecciona para el ejemplo de osciladores cúbicos (O.C), mostrado en el Capítulo siguiente, el valor $\lambda = 0,09$ con el cual se reconstruyen las ecuaciones dinámicas para un valor del paso de integración $dt = 0,0001$.

Algoritmo de aprendizaje automático: SINDy

En este capítulo se realiza una revisión general del problema sobre la reconstrucción de modelos *parsimonios* para sistemas dinámicos; usando series temporales e implementando herramientas de aprendizaje automático (Machine Learning) se conforma un algoritmo para obtener soluciones dispersas, involucrando técnicas de regularización y optimización como las descritas en el Capítulo 5, haciendo que las ecuaciones gobernantes sean dispersas en un espacio de funciones no lineales multidimensional. Existen en la literatura trabajos recientes ([19], [20]) que usan combinación de diferentes métodos, como regresión dispersa, para encontrar ecuaciones dinámicas.

6.1. SINDy

El algoritmo de aprendizaje automático (Machine Learning) denominado como SINDy (Sparse Identification of Non-Linear Dynamics) [16], basado en regresión dispersa junto con técnicas de regularización y optimización, permite construir modelos dinámicos no lineales a partir de bases de datos (series temporales) que han sido recolectadas por simulaciones o experimentos. Representando un sistema dinámico no lineal como en (4.4) de la forma

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}(t)) , \quad (6.1)$$

que es una ecuación diferencial ordinaria de primer orden de n variables de estado donde se define el vector columna

$$\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T \in \mathbb{R}^n , \quad (6.2)$$

que representa el estado del sistema a un tiempo t y la función $\mathbf{f}(\mathbf{x}(t))$ representa las restricciones dinámicas, como la rigidez, amortiguamiento, fuerza de fricción o fuerzas externas, que definen las ecuaciones de movimiento.

Para determinar $\mathbf{f}(\mathbf{x}(t))$ se hace uso de la evolución temporal en las variables de estado en $\mathbf{x}(t)$ y las derivadas $\dot{\mathbf{x}}(t)$, las cuales se pueden calcular de forma aproximada a partir de métodos como el de diferencias finitas, Savitzky Golay, etc. Así, las muestras de datos recolectados a diferentes instantes en el tiempo $t_1, t_2, \dots, t_m$, pueden ser escritos de forma matricial [16] como

$$\mathbf{X}(t) = \begin{bmatrix} \mathbf{x}^T(t_1) \\ \mathbf{x}^T(t_2) \\ \vdots \\ \mathbf{x}^T(t_m) \end{bmatrix} = \begin{bmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & \cdots & x_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t_m) & x_2(t_m) & \cdots & x_n(t_m) \end{bmatrix}, \quad (6.3)$$

$$\dot{\mathbf{X}}(t) = \begin{bmatrix} \dot{\mathbf{x}}^T(t_1) \\ \dot{\mathbf{x}}^T(t_2) \\ \vdots \\ \dot{\mathbf{x}}^T(t_m) \end{bmatrix} = \begin{bmatrix} \dot{x}_1(t_1) & \dot{x}_2(t_1) & \cdots & \dot{x}_n(t_1) \\ \dot{x}_1(t_2) & \dot{x}_2(t_2) & \cdots & \dot{x}_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \dot{x}_1(t_m) & \dot{x}_2(t_m) & \cdots & \dot{x}_n(t_m) \end{bmatrix}, \quad (6.4)$$

se propone una librería $\Theta(\mathbf{X})$ de funciones lineales o no-lineales con las que se podría llegar a representar la dinámica de las series temporales, estas funciones son especificadas por el usuario y se pueden utilizar términos constantes, polinomiales, trigonométricos, exponenciales, etc. , por ejemplo:

$$\Theta(\mathbf{X}) = \begin{bmatrix} 1 & \mathbf{X} & \mathbf{X}^{P_2} & \mathbf{X}^{P_3} & \cdots & \sin(\mathbf{X}) & \cos(\mathbf{X}) & \sin(2\mathbf{X}) & \cos(2\mathbf{X}) & \cdots \end{bmatrix}. \quad (6.5)$$

Los polinomios de orden mayor se denotan como $\mathbf{X}^{P_2}, \mathbf{X}^{P_3}$, etc. Por ejemplo, para $\mathbf{X}^{P_2}$ la cual contiene las no linealidades cuadráticas en el estado del sistema , se tiene

$$\mathbf{X}^{P_2} = \begin{bmatrix} x_1^2(t_1) & x_1(t_1)x_2(t_1) & \cdots & x_2^2(t_1) & x_2(t_1)x_3(t_1) & \cdots & x_n^2(t_1) \\ x_1^2(t_2) & x_1(t_2)x_2(t_2) & \cdots & x_2^2(t_2) & x_2(t_2)x_3(t_2) & \cdots & x_n^2(t_2) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_1^2(t_m) & x_1(t_m)x_2(t_m) & \cdots & x_2^2(t_m) & x_2(t_m)x_3(t_m) & \cdots & x_n^2(t_m) \end{bmatrix}. \quad (6.6)$$

Entonces tenemos que cada elemento de $\Theta(\mathbf{X})$ representa una función candidata para el lado derecho en (6.1). Se considera que la librería $\Theta(\mathbf{X})$ es una matriz con dimensiones $m \times p$, donde p es el número de funciones candidatas no lineales y m es el número total de muestras de datos; entre más datos se manejen para la obtención de un modelo se cumple que $m > p$, por lo tanto, el sistema es sobredeterminado.

En un sistema dinámico no todos los elementos de la librería estarán activos, ya que existen variables independientes que no influyen en el modelo, por lo tanto se llega a una solución dispersa. Esto se logra encontrando un vector columna de pesos $\Xi = [\xi_1 \ \xi_2 \ \dots \ \xi_n]^T$, donde cada elemento $\xi_i \neq 0$ ($i = 1, \dots, n$) nos proporciona los términos activos en $\Theta(\mathbf{X})$, los cuales representan al modelo que se ajusta a la dinámica de la serie temporal. Por lo tanto el problema de dispersión para encontrar las ecuaciones del modelo se obtiene con Ξ en el espacio de estados y es de la forma

$$\dot{\mathbf{X}} = \Theta(\mathbf{X})\Xi. \quad (6.7)$$

El problema de dispersión se resuelve con el método de regresión lineal por mínimos cuadrados,

$$\xi_k = \arg \min_{\xi_k} \left\| \dot{\mathbf{X}}_k - \Theta(\mathbf{X})\xi_k \right\|^2, \quad (6.8)$$

encontrando así los pesos de Ξ , que junto con técnicas de regularización (Sección 5.3) se promueve que el vector sea disperso. Con técnicas de optimización el modelo final es parsimonioso. Una vez calculada Ξ , se forma una ecuación dinámica para cada variable de estado de la forma

$$\dot{\mathbf{x}}_k = \mathbf{f}_k(\mathbf{x}) = \Theta(\mathbf{x}^T)\xi_k, \quad (6.9)$$

en conjunto para todas las variables de estado se obtiene la ecuación,

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) = \Xi^T(\Theta(\mathbf{x}^T))^T, \quad (6.10)$$

que representa el modelo obtenido por el algoritmo. Dados los datos en la matriz $\mathbf{X}$, se calculan para (6.7) las derivadas $\dot{\mathbf{X}}$ de forma numérica. En el caso que los datos contengan ruido las derivadas también estarán contaminadas, entonces (6.7) debe modificarse

de la forma

$$\dot{\mathbf{X}} = \mathbf{\Theta}(\mathbf{X})\mathbf{\Xi} + \eta\mathbf{Z} , \quad (6.11)$$

donde $\mathbf{Z}$ es una matriz de entradas gaussianas independientes distribuidas de forma idéntica con una media cero y η es la magnitud del ruido, así que en el caso general se busca una solución dispersa para un sistema sobredeterminado con ruido.

La solución de la ecuación (6.7) se resume entonces en la obtención de un modelo parsimonio desde el enfoque del aprendizaje automático supervisado usando regresión lineal, donde se calculan los pesos de las funciones de la librería $\mathbf{\Theta}(\mathbf{X})$ mediante mínimos cuadrados; usando técnicas de regularización (por ejemplo (5.45), STLQ, etc.), se penalizan los pesos con el umbral del hiperparámetro λ cuyo valor óptimo se obtiene con la técnica de validación cruzada frente de Pareto, encontrando modelos para cada valor de los hiperparámetros, estableciendo al final el modelo que minimice el error entre los datos recolectados y simulados. Este procedimiento se puede extender para la solución con ecuaciones diferenciales parciales sustituyendo el planteamiento en (6.1). El proceso para obtener un modelo parsimonio usando series de datos mediante SINDy se puede ilustrar en la Figura 6.1.

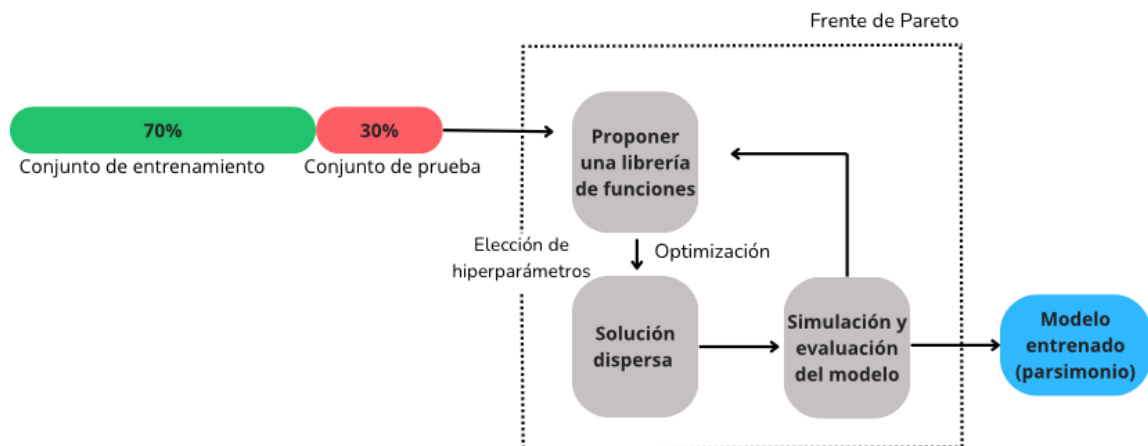


Figura 6.1: Representación del proceso para la obtención de modelos parsimonios con el algoritmo SINDy, usando la técnica frente de Pareto en el cual se comparan de forma cuantitativa los modelos obtenidos con diferentes valores de los hiperparámetros de la regularización.

6.2. Paquetería PySINDy

Para la aplicación del algoritmo SINDy se hace uso de una paquetería disponible en Python denominada PySINDy. En [21] se describen un par de esquemas en los cuales se muestran la opciones que ofrece la paquetería (Fig. 6.2, Fig. 6.3) actualizadas hasta el año 2022, primero para la identificación de las ecuaciones gobernantes en un sistema no lineal y un resumen de ayuda al usuario con objetivo de realizar una buena elección de las librerías candidatas junto con un optimizador.

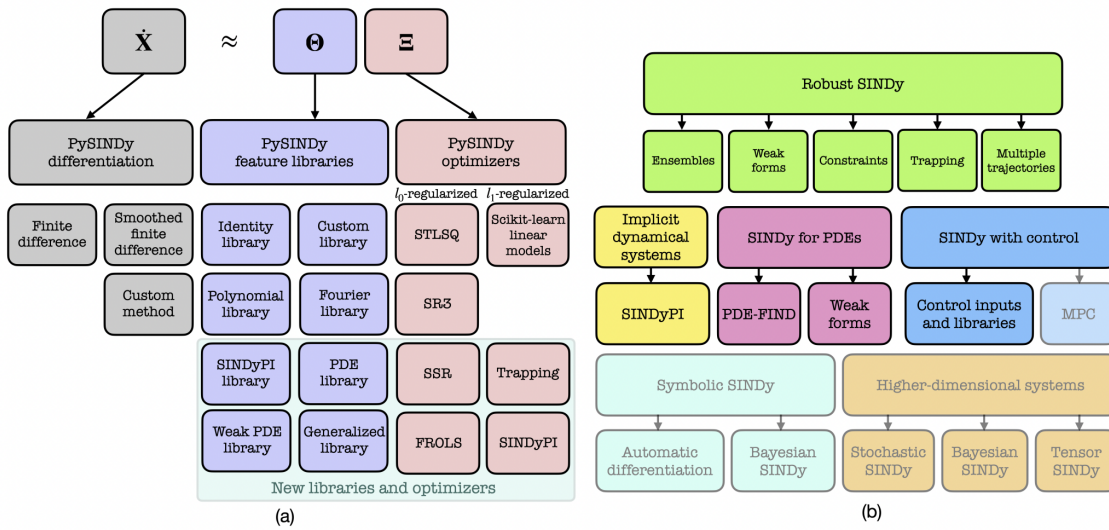


Figura 6.2: Resumen de características, tomado de [21], organizado por (a) Estructura de PySINDy y (b) funcionalidad. En (a) se muestra la jerarquía desde el problema de regresión dispersa resuelto por SINDy, hasta los submódulos de PySINDy, a los optimizadores individuales, bibliotecas y métodos de diferenciación implementados en el código. (b) es un diagrama para las variantes de SINDy y funcionalidades descritas en la literatura. Los recuadros transparentes corresponden a características que no se han implementado aún.

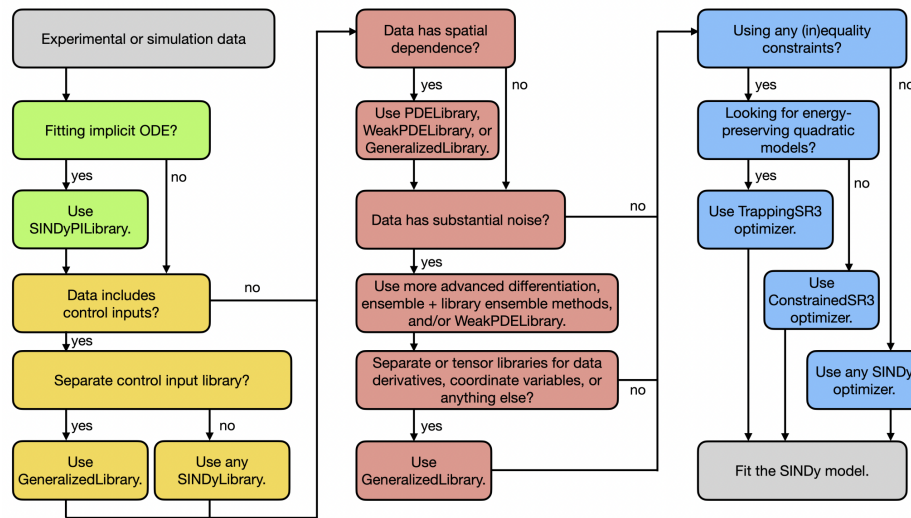


Figura 6.3: Diagrama, tomado de [21], que resume como el usuario puede comenzar con una base de datos y sistemáticamente seleccionar debidamente las librerías candidatas junto con el optimizador de regresión dispersa que se adaptan a un problema en específico.

6.3. Reconstrucción de ecuaciones dinámicas con PySINDy: Ejemplos

Como ilustración de la aplicación de SINDy se muestran los ejemplos mostrados en el anexo de [16], donde se conocen las ecuaciones dinámicas de los sistemas, lo cual hace trivial la elección de una librería candidata para la reconstrucción del modelo. En cada ejemplo se obtienen las series temporales a usar en SINDy, integrando las ecuaciones originales utilizando una extensión del método Runge-Kutta de orden 4 (Sec. 4.4.2), denominado RK-Fehlberg (RKF-45) [12]. Para el proceso de regularización de los siguientes ejemplos se hace uso de los métodos STLQ, validación cruzada y frente de pareto (usando 70 % de datos para el entrenamiento y 30 % para la prueba del modelo obtenido) descritos en la Sección 5.3, ajustando así el valor del hiperparámetro λ (el cual limita el número de términos en los modelos construidos) para la obtención de modelos parsimonios, los cuales deben ser iguales a las ecuaciones dinámicas ya conocidas.

En estos ejemplos y el desarrollo del trabajo más adelante, las series temporales y/o sus derivadas no contienen ruido; si se desea conocer la implementación de SINDy

para datos con ruido, en específico de los ejemplos aquí ilustrados, se recomienda al lector revisar [22].

6.3.1. Oscilador lineal y cúbico

Se presentan como ejemplos ilustrativos del algoritmo SINDy algunos sistemas de los cuales se reconstruyen las ecuaciones dinámicas. El primer sistema de ecuaciones en (6.12), (6.13) describe la dinámica de un oscilador con dos grados de libertad, de osciladores armónicos lineales (O.L) acoplados que pueden presentar amortiguamiento dependiendo el valor de los parámetros a , b , c , d (ejemplo tomado del Anexo de [16]), donde las variables de estado $x_1(t)$, $x_2(t)$ representan los desplazamientos de dos masas,

$$\frac{d}{dt}x_1(t) = -ax_1 + bx_2, \quad (6.12)$$

$$\frac{d}{dt}x_2(t) = cx_1 + dx_2, \quad (6.13)$$

y a, b, c, d son parámetros del sistema, mediante los cuales se exhibirá una diversidad de comportamientos dinámicos dependiendo del valor que tomen y las condiciones iniciales que se utilicen. El análisis detallado de la estabilidad y las soluciones del sistema puede llevarse a cabo con técnicas de teoría en sistemas dinámicos. En particular para este ejemplo se consideran los valores de los parámetros $a = -0,1, b = 2, c = -2, d = -0,1$, la reconstrucción de las ecuaciones se realiza con diferentes valores de la separación temporal entre los datos dt (que también es el tamaño del paso de integración numérica) y con valor de $\lambda = 0,05$, determinado usando frente de Pareto, Fig. 6.4 . Donde se observa que al incrementar dicho valor, visto de forma cuantitativa, el error cuadrático medio (MSE) incrementa de forma considerable para los valores $dt = 0,1$ y $0,25$, el coeficiente de correlación R^2 disminuye y los coeficientes de las ecuaciones se alejan del valor exacto, indicando así una pérdida de precisión en la reconstrucción del modelo exacto, Tabla 6.1. Se muestran para un par de casos las series temporales y espacio fase en la Fig. 6.6.

Igualmente, usando la misma metodología del ejemplo anterior, se ilustra la reconstrucción con el sistema de ecuaciones para un oscilador armónico no-lineal con términos

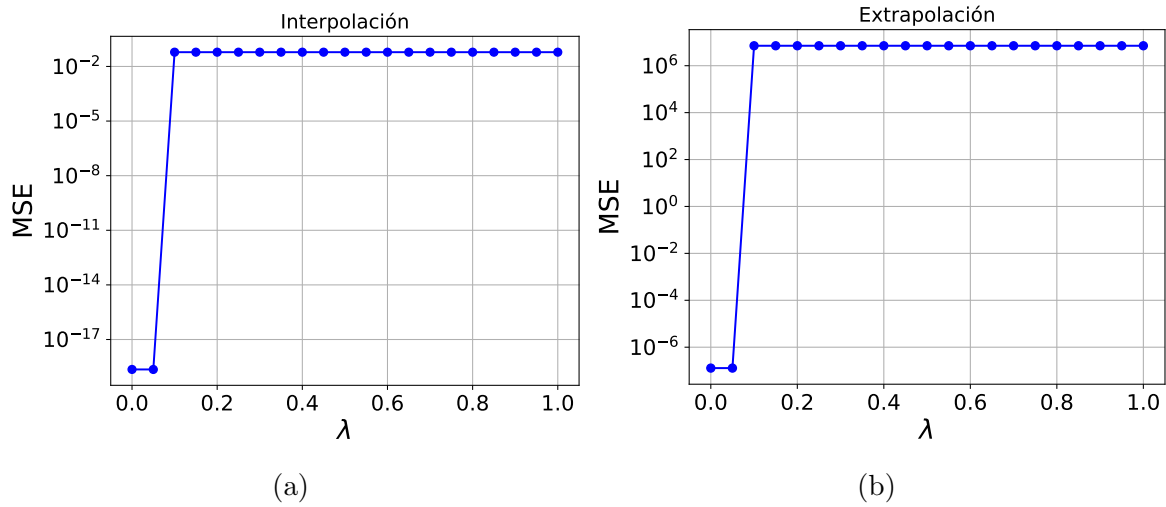


Figura 6.4: Gráficas en escala logarítmica de la interpolación (a) y extrapolación (b) de las series de tiempo, donde se compara el error cuadrático medio (MSE) de los modelos obtenidos con diferentes valores del hiperparámetro λ en la regularización con STLQ. Se selecciona para el ejemplo de osciladores lineales (O.L) el valor $\lambda = 0,05$ con el cual se reconstruyen las ecuaciones dinámicas para un valor del paso de integración de $dt = 0,0001$. Se realizó el mismo análisis para todos los valores del parámetro dt en el Cuadro 6.1.

cúbicos (O.C),

$$\frac{d}{dt}x_1(t) = Ax_1^3 + Bx_2^3, \quad (6.14)$$

$$\frac{d}{dt}x_2(t) = Cx_1^3 + Dx_2^3; \quad (6.15)$$

como en el sistema con términos lineales, la diversidad en la dinámica se encuentra en los parámetros de las ecuaciones, aquí se considera $A = -0,1, B = 2, C = -2, D = -0,1$, con los cuales se obtiene la reconstrucción de la Tabla (6.2) usando diferentes valores en dt junto con las correspondientes series temporales y espacio fase en Fig. 6.7.

6.3.2. Péndulo Simple

Ahora pasemos a un ejemplo más complejo: el péndulo simple. Este sistema se conforma por una partícula puntual con una masa suspendida por un hilo inextensible y de masa despreciable. Al ser desplazada de su posición de equilibrio, la fuerza gravita-

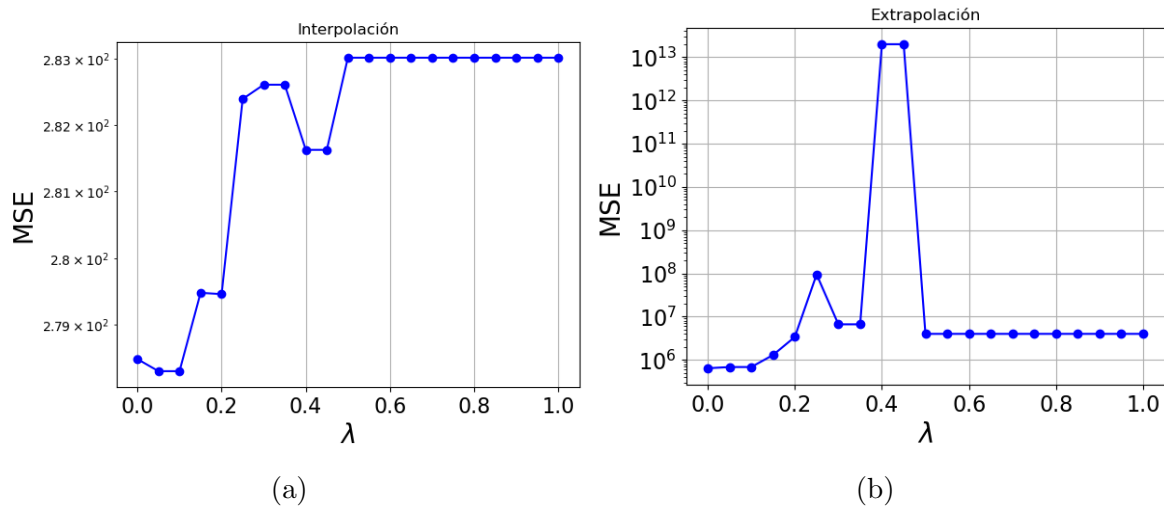


Figura 6.5: Gráficas en escala logarítmica de la interpolación (a) y extrapolación (b) de las series de tiempo, donde se compara el error cuadrático medio MSE de los modelos obtenidos con diferentes valores del hiperparámetro λ en la regularización con STLQ. Se selecciona para el ejemplo de osciladores cúbicos (O.C) el valor $\lambda = 0,09$ con el cual se reconstruyen las ecuaciones dinámicas para un valor del paso de integración $dt = 0,0001$. Se realizó el mismo análisis para todos los valores del parámetro dt en el Cuadro 6.2.

dt	Modelo SINDy	MSE	R^2
0.0001	$\dot{x}_1 = -0,1x_1 + 1,9999999x_2$ $\dot{x}_2 = -1,9999999x_1 - 0,1x_2$	$5,250 \times 10^{-14}$	0.999999999999
0.001	$\dot{x}_1 = -0,0999998x_1 + 1,99999868x_2$ $\dot{x}_2 = -1,99999868x_1 - 0,0999998x_2$	$5,244 \times 10^{-10}$	0.99999999994
0.01	$\dot{x}_1 = -0,09998027x_1 + 1,99986774x_2$ $\dot{x}_2 = -1,99986934x_1 - 0,09998013x_2$	$5,179 \times 10^{-6}$	0.999992967025
0.1	$\dot{x}_1 = -0,09841235x_1 + 1,9868566x_2$ $\dot{x}_2 = -1,98838231x_1 - 0,09809546x_2$	0.0461	0.995237255
0.25	$\dot{x}_2 = -0,097x_1 + 1,918x_2$ $\dot{x}_2 = -1,936x_1$	4.0087	0.376770429

Cuadro 6.1: Modelos obtenidos para el sistema de osciladores lineales, cada uno con un valor particular de λ .

cional hace efecto induciendo un patrón oscilatorio periódico alrededor de su punto de equilibrio. La dinámica del péndulo simple se describe por las ecuaciones no lineales de

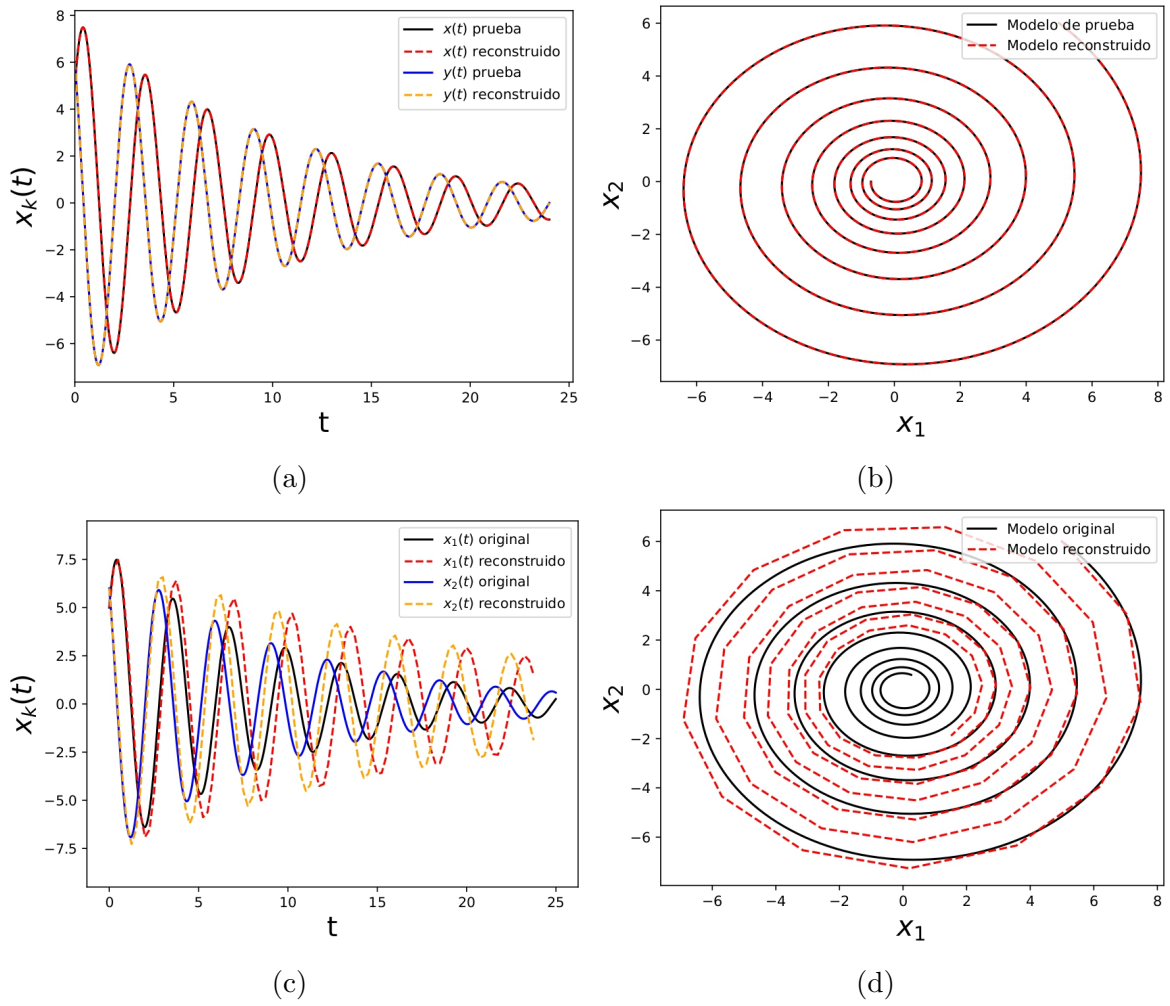


Figura 6.6: Serie temporal y espacio fase para la reconstrucción de las ecuaciones (6.12), (6.13) con $dt = 0,0001$ en (a), (b), donde se recupera el modelo exacto y para $dt = 0,25$ en (c), (d), donde se pierde la precisión del modelo.

primer orden

$$\frac{d}{dt}x_1(t) = x_2, \quad (6.16)$$

$$\frac{d}{dt}x_2(t) = -\frac{g}{L} \sin x_1, \quad (6.17)$$

donde aparece una función trigonométrica con el ángulo denotado por la variable x_1 .

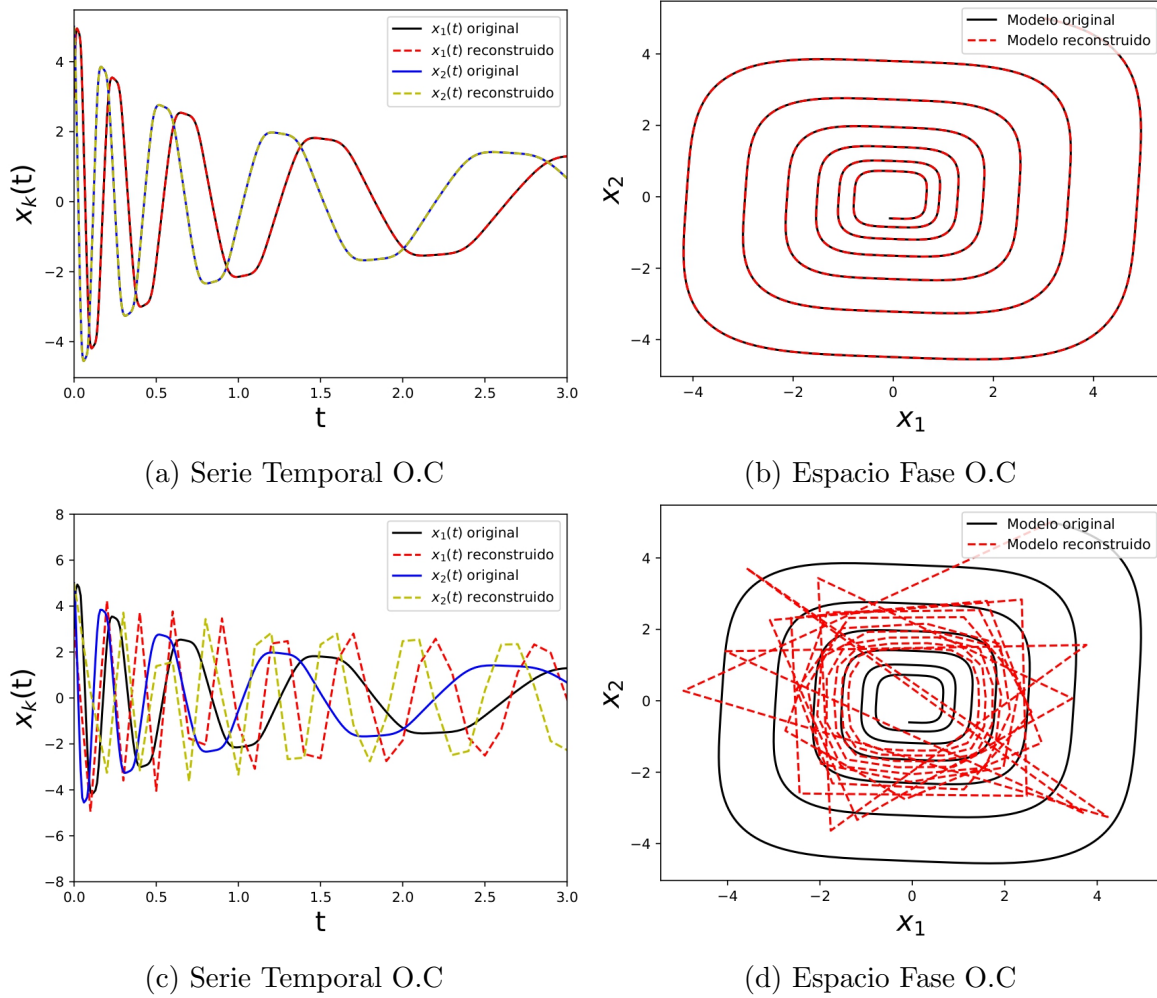


Figura 6.7: Serie temporal y espacio fase para la reconstrucción de las ecuaciones (6.14), (6.15) con $dt = 0,0001$ en (a), (b) y $dt = 0,1$ en (c), (d).

El sistema puede ser aproximado para pequeñas amplitudes, expandiendo el término $\sin x_1$ como:

$$\sin x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots ; \quad (6.18)$$

dt	Modelo SINDy	MSE	R^2
0.0001	$\dot{x} = -0,10000004x^3 + 1,99999988y^3$ $\dot{y} = -1,99999988x^3 - 0,09999995y^3$	$3,509 \times 10^{-13}$	0.9999999999
0.001	$\dot{x} = -0,10000363x^3 + 1,99998805y^3$ $\dot{y} = -1,99998842x^3 - 0,09999512y^3$	$3,412 \times 10^{-9}$	0.9999999641
0.01	$\dot{x} = -0,10030659x^3 + 1,99880939y^3$ $\dot{y} = -1,99886234x^3 - 0,09951372y^3$	$6,723 \times 10^{-6}$	0.99999296702
0.1	$\dot{x} = 0,19646535x^2y + 1,84753065y^3$ $\dot{y} = -0,12231772x^2 - 2,02903184x^3$ $-0,15985548x^2y$	3.270	-2.168

Cuadro 6.2: Modelos obtenidos para el sistema de osciladores cúbicos, cada uno con un valor particular de λ .

por lo tanto se puede escribir el conjunto de ecuaciones de forma lineal

$$\frac{d}{dt}x_1(t) = x_2, \quad (6.19)$$

$$\frac{d}{dt}x_2(t) = -\frac{g}{L}x_1. \quad (6.20)$$

Aunque las ecuaciones dinámicas del péndulo se pueden aproximar de forma lineal con las ecuaciones (6.19) y (6.20), en este ejemplo se utiliza la representación no-lineal con (6.16), (6.17) y los parámetros $g = 9,81$, $L = 1$, para mostrar la habilidad de SINDy al combinar librerías de diferentes tipos de funciones. En este caso se utilizan librerías de funciones polinomiales y de Fourier para la reconstrucción del modelo igual que los ejemplos anteriores se realizó el análisis con frente de Pareto para determinar el valor de λ óptimo para cada caso dt . Los resultados se muestran en la Tabla 6.3 para diferentes valores en dt . Las gráficas de las series temporales junto con el espacio fase para $dt = 0,0001$ se muestran en Fig. 6.8.

6.4. Atractor de Lorenz

El siguiente sistema dinámico no lineal surge de un modelo simplificado de convección atmosférica obtenido por Edward Lorenz en 1963 [23]. Este sistema se caracteriza

dt	Modelo SINDy	MSE	R^2
0.0001	$\dot{x} = 0,999999999y$ $\dot{y} = -9,80999986 \sin x$	$3,358 \times 10^{-12}$	0.9999999999
0.001	$\dot{x} = 0,99999902x$ $\dot{y} = -9,80998644 \sin x$	$3,316 \times 10^{-8}$	0.99999998458
0.01	$\dot{x} = 0,99990216x$ $\dot{y} = -9,80864561 \sin x$	$3,312 \times 10^{-4}$	0.99984605662
0.1	$\dot{x} = 0,99033839x$ $\dot{y} = -0,45519266x - 8,91688847 \sin x$	14270.5	-34.49

Cuadro 6.3: Modelos obtenidos para el sistema del péndulo simple, cada uno con un valor particular de λ .

por el conjunto de tres ecuaciones de primer orden,

$$\frac{dx}{dt} = \sigma(y - x) \quad (6.21)$$

$$\frac{dy}{dt} = x(\rho - z) - y \quad (6.22)$$

$$\frac{dz}{dt} = xy - \beta z, \quad (6.23)$$

donde σ , ρ , β son parámetros del sistema que corresponden al número de Prandtl (razón entre viscosidad y conductividad térmica), número de Rayleigh (relaciona la diferencia de temperaturas mínima y máxima del sistema, la conductividad térmica, la capacidad calorífica y la viscosidad), junto con la razón entre la longitud y altura del sistema, respectivamente. Usando SINDy se realiza la reconstrucción, para diferentes valores en dt (seleccionando λ como se realizó en ejemplos anteriores), de las ecuaciones (6.21)-(6.23) con los valores de $\sigma = 10$, $\rho = 28$, $\beta = 8/3$, donde el modelo exhibe un comportamiento caótico.

Lo que resulta de la reconstrucción del modelo en el Cuadro 6.4, de forma cuantitativa, pareciera ser no satisfactorio debido a que el error cuadrático medio es del orden $\approx 10^8$ en comparación con lo que se obtenía en ejemplos anteriores, donde el error entre las series temporales de las ecuaciones reconstruidas y las originales era del orden $\approx 10^{-10}$ para valores dt infinitesimales. En términos del coeficiente de correlación R^2 , evaluando los modelos se obtienen valores negativos que concuerdan con el intervalo $R \in [-1, 1]$, pero indican falta de precisión al ser > -1 ó < -1 .

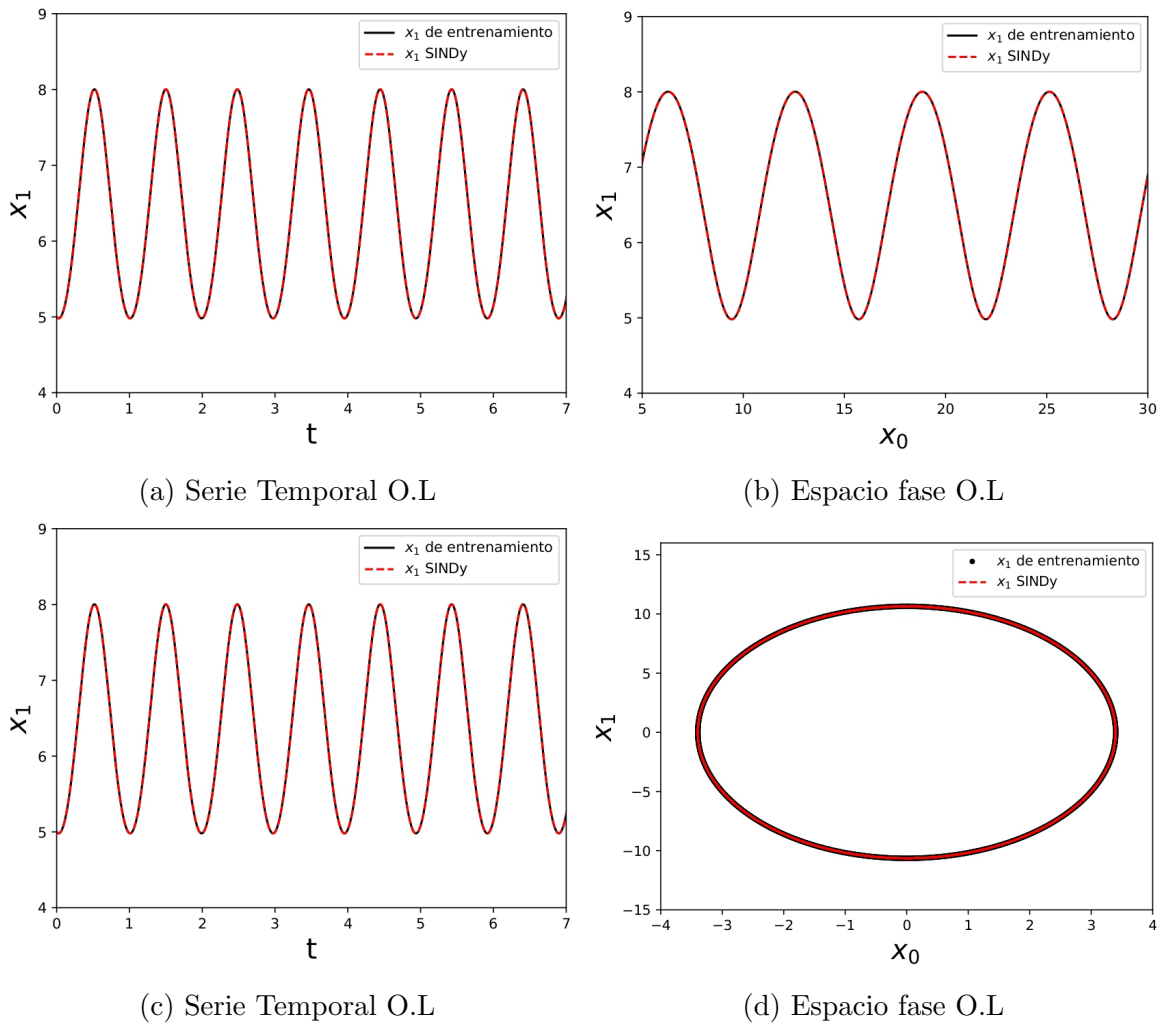


Figura 6.8: Representación del sistema en coordenadas polares (a) y (b), y la representación lineal (c), (d) en el sistema del péndulo simple.

Pero cabe resaltar que a pesar de esto, para algunos valores en dt , como el caso $dt = 1 \times 10^{-4}$ ó $dt = 0,005$, los términos de las ecuaciones (6.21)-(6.23) son recuperados con valores de los coeficientes cercanos a los esperados. Comparando las series temporales, por ejemplo de la variable y mostrada en la Fig. 6.9 (a), en el caso $dt = 0,0001$, se observa que al inicio de la evolución temporal del sistema, el modelo obtenido reproduce en buena medida el comportamiento de la serie temporal original sin embargo, después de cierto tiempo, se evidencia una desviación gradual entre el modelo y la serie temporal original, lo que sugiere la aparición de discrepancias significativas en la capacidad

dt	Modelo SINDy	MSE	R^2
0.0001	$\dot{x} = -9,99999785x + 9,99999785y$ $\dot{y} = 27,99998004x - 0,99999612y$ $\quad -0,99999944xz$ $\dot{z} = -2,66666924z + 0,99999971xy$	101,979957	-0.42016
0.005	$\dot{x} = -9,99455541x + 9,99454987y$ $\dot{y} = 27,95068046x - 0,99049541y$ $\quad -0,99861659xz$ $\dot{z} = -2,66472839z + 0,99927128xy$	123,48844	-0.74481
0.1	$\dot{x} = 0,9811 - 15,984x$ $\quad +13,922y + 0,239xz$ $\quad -0,189yz$ $\dot{y} = 0,8351 + 14,204x$ $\quad +1,297y - 0,596xz$ $\dot{z} = 30,1811 + 0,236x$ $\quad -3,929z + 0,685x^2$ $\quad +0,235y^2$	143,767244	-1.10967

Cuadro 6.4: Modelos obtenidos para el sistema del atractor de Lorenz, cada uno con un valor particular de λ .

predictiva del modelo. Este fenómeno puede ser atribuido a la naturaleza caótica del sistema.

Todo esto sugiere que el algoritmo SINDy experimenta una pérdida de precisión en la reproducción del comportamiento cuando se aplica a sistemas caóticos. Por lo tanto se propone investigar en el siguiente capítulo, mediante otro sistema caótico estudiado por primera vez con herramientas de aprendizaje automático la variación del error relativo, entre los coeficientes obtenidos y los originales, a medida que se incrementa la separación entre los datos de las series temporales (paso de integración), así como explorar el comportamiento del algoritmo cuando los sistemas dependen de parámetros, como la energía o propios parámetros en las ecuaciones dinámicas. Además surge una interrogante que abre una línea de investigación para trabajos futuros: Utilizando técnicas de aprendizaje automático, ¿Es posible mitigar la pérdida de precisión a tiempos largos ante sistemas que presentan caos?.

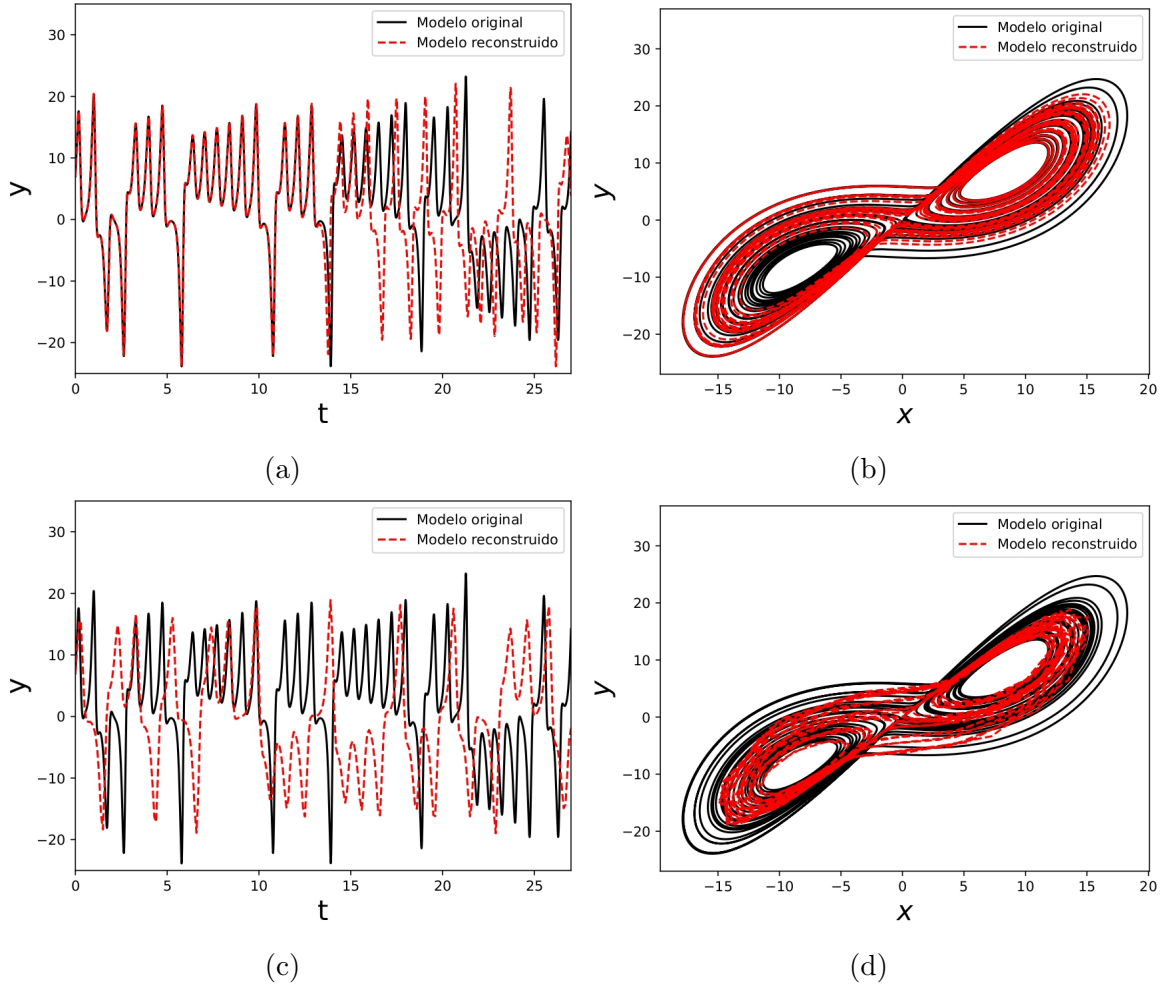


Figura 6.9: Serie temporal de la variable y junto con los planos x vs y para los valores $dt = 1 \times 10^{-4}$ en (a),(b) y $dt = 5 \times 10^{-3}$ en (c),(d). Usando valores infinitesimales en dt el algoritmo recupera la estructura de las ecuaciones dinámicas exactas pero pierde la capacidad de reproducir la órbita en cierto valor del tiempo, como se muestra en (a), (b); usando valores ligeramente más grandes del paso de integración, se reconstruyen las ecuaciones esperadas pero no la dinámica, como se muestra en (c) y (d).

Sistema generalizado de Hénon-Heiles

A lo largo de este capítulo, se discute la reconstrucción de las ecuaciones de Hamilton para la generalización del sistema original de Hénon-Heiles donde el potencial es la suma de un oscilador armónico isotrópico bidimensional y una función polinomial de grado $N = 1, 2, 3, \dots$ en las variables (x, y) . Esta generalización es utilizada como prototipo de prueba del algoritmo SINDy ante la presencia de caos como función de la energía y el parámetro N .

7.1. Sistema de Hénon-Heiles

El potencial de Hénon-Heiles [24] fue resultado de la investigación por parte de la comunidad astronómica con interés en describir el movimiento de una estrella alrededor de un campo gravitacional de una galaxia axisimétrica. Este sistema clásico no-relativista con dos grados de libertad, está descrito por el Hamiltoniano

$$H = \frac{1}{2}(p_x^2 + p_y^2) + \frac{1}{2}(x^2 + y^2) + g(yx^2 - \frac{1}{3}y^3) , \quad (7.1)$$

con el parámetro real $g > 0$ como una constante de acoplamiento, donde (x, y) son coordenadas cartesianas y p_x, p_y sus respectivos momentos canónicos. La energía total $H = E$ del sistema en dos dimensiones es la única cantidad conservada. El potencial en (7.1) se puede ver como un oscilador armónico isotrópico bidimensional con una perturbación de términos cúbicos no lineales. Este Hamiltoniano incorpora la propiedad física de que (I) la estrella puede ser atrapada bajo dicho potencial y (II) aquellas

que contienen suficiente energía pueden escapar de la galaxia que orbitan. El análisis numérico realizado en el trabajo original muestra esta propiedad, ya que se revela la existencia de un valor crítico de energía E_c el cual representa una cota para la energía de escape de la estrella. Para energías pequeñas $E \ll E_c$ las órbitas no son influenciadas de forma significativa por los términos cúbicos del potencial y se tienen en su mayoría comportamientos periódicos. Para el caso en que $E \sim E_c$ la presencia de trayectorias caóticas es relevante, en cuestiones del espacio fase, casi todos los toros invariantes se rompen. Existen trabajos relacionados a la generalización de este potencial (7.1), considerando por ejemplo los términos cúbicos con parámetros libres o agregando términos de orden mayor,[25],[26]. También se encuentran trabajos que exploran la integrabilidad de sistemas tipo Hénon-Heiles como en [27].

7.1.1. Potencial de Hénon Heiles

En este sistema con dos grados de libertad se considera que un cuerpo está bajo la presencia de un potencial de la forma

$$V(x, y) = \frac{x^2 + y^2}{2} + x^2y - \frac{y^3}{3} \quad (7.2)$$

el cual representa en el primer término dos osciladores armónicos simples pero con una perturbación que los acopla, este término no lineal es $x^2y - \frac{y^3}{3}$. Dicho potencial se presenta gráficamente a continuación. En este tipo de sistemas conservativos, se realiza un análisis del espacio fase identificando los puntos fijos y así estudiar la estabilidad de las trayectorias cerca de estos punto; pueden existir puntos que son elípticos o hiperbólicos, estos últimos son asociados a trayectorias en el espacio fase que pueden estar acercándose o alejándose del punto fijo y se hacen inestables. En los elípticos las trayectorias rotan alrededor de los puntos. Se encuentran los puntos fijos de forma analítica, considerando las ecuaciones de movimiento presentadas en (7.9)-(7.12), haciendo $\dot{x} = \dot{y} = \dot{p}_x = \dot{p}_y = 0$, por lo que se obtienen las siguientes ecuaciones:

$$\begin{aligned} p_x &= 0 , \\ p_y &= 0 , \\ x(1 + 2y) &= 0 , \\ y + x^2 - y^2 &= 0 . \end{aligned} \quad (7.3)$$

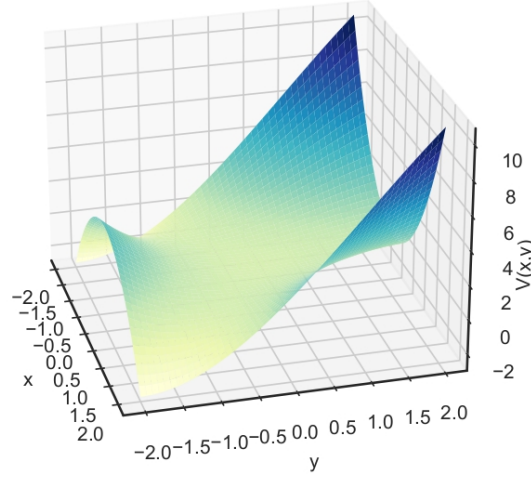


Figura 7.1: Gráfica del potencial $V(x, y)$ en (7.2) de Hénon Heiles.

Al resolver el sistema de ecuaciones en (7.3), se encuentran los puntos fijos $P(x, y, p_x, p_y)$:

$$P_1 = (0, 0, 0, 0) , \quad (7.4)$$

$$P_2 = (0, 1, 0, 0) , \quad (7.5)$$

$$P_3 = \left(-\frac{\sqrt{3}}{2}, -\frac{1}{2}, 0, 0\right) , \quad (7.6)$$

$$P_4 = \left(\frac{\sqrt{3}}{2}, \frac{1}{2}, 0, 0\right) . \quad (7.7)$$

Considerando diferentes planos del potencial (7.2) correspondientes a diferentes valores de la energía E , se obtienen las curvas equipotenciales. En la Fig. 7.2, se observa que al considerar una energía $E = \frac{1}{6}$, la curva equipotencial se convierte en tres líneas que forman un triángulo definido por los puntos de equilibrio en (7.4)-(7.7), por lo que se presenta un movimiento acotado dentro del triángulo y al considerar $E < 1/6$ se tienen trayectorias acotadas. Entonces la energía crítica E_c del sistema se puede considerar como $E = 1/6$.

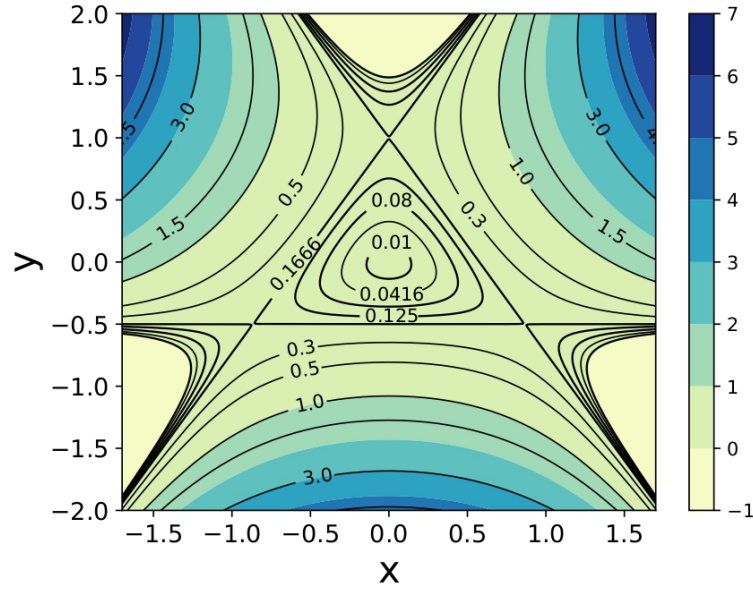


Figura 7.2: Curvas de nivel para el sistema de Hénon Heiles para energías $E < 1/6$ (contornos dentro del triángulo de energía crítica) y $E > 1/6$ (contornos fuera de la región de energía crítica).

7.1.2. Ecuaciones de movimiento

A partir del Hamiltoniano para el sistema de Hénon Heiles,

$$H = \frac{p_x^2 + p_y^2}{2} + \frac{x^2 + y^2}{2} + x^2y - \frac{y^3}{3}, \quad (7.8)$$

se obtienen las ecuaciones de Hamilton para cada variable dinámica;

$$\dot{x} = \frac{\partial H}{\partial p_x},$$

$$\dot{y} = \frac{\partial H}{\partial p_y},$$

$$\dot{p}_x = -\frac{\partial H}{\partial x},$$

$$\dot{p}_y = -\frac{\partial H}{\partial y},$$

explícitamente:

$$\dot{x} = p_x , \quad (7.9)$$

$$\dot{y} = p_y , \quad (7.10)$$

$$\dot{p}_x = -x - 2xy , \quad (7.11)$$

$$\dot{p}_y = -y - x^2 + y^2 . \quad (7.12)$$

A manera de ilustración se resuelven estas ecuaciones diferenciales, que contienen términos no lineales usando el método de integración numérica Runge-Kutta de orden 4 descrito en el Capítulo 4.4; se muestran las series temporales obtenidas para una condición inicial aleatoria en la Fig. 7.3.

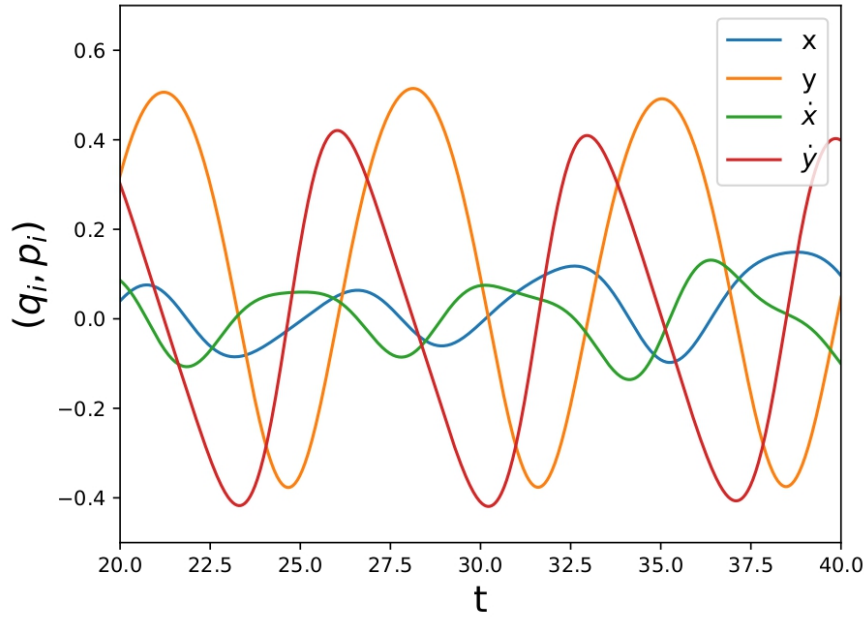


Figura 7.3: Solución numérica de las ecuaciones canónicas para una condición inicial aleatoria $(x_0, y_0, p_{x_0}, p_{y_0}) = (0, 0, 1, 0, 470177, 0, 14)$.

7.2. Sistema Generalizado de Hénon Heiles

El estudio que se presenta aquí considera una generalización del potencial (7.2), donde el nuevo potencial es la suma de un oscilador armónico isotrópico bidimensional y una función polinomial de grado $N = 1, 2, 3, \dots$ en las variables (x, y) . El sistema

generalizado admite una simetría dihedral D_N (grupo de simetrías de un polígono regular con N aristas) y como se mencionó anteriormente, existe un valor de energía crítica $E_c = E_c(N)$ el cual separa los movimientos acotados de los no acotados.

Dicha generalización se escribe de la forma

$$\mathcal{H}_N = \frac{1}{2m} (p_r^2 + \frac{1}{r^2} p_\theta^2) + V_N(r, \theta), \quad (7.13)$$

con potencial escalar

$$V_N(r, \theta) = \frac{1}{2} m \omega^2 r^2 + \frac{1}{N} r^N \sin(N\theta), \quad (7.14)$$

donde $r = \sqrt{x^2 + y^2}$, $\theta = \tan^{-1}(y/x)$ son coordenadas polares, p_r , p_θ los momentos conjugados respectivos, m es la masa de la partícula, $\omega > 0$ una frecuencia y $N > 0$ un número entero positivo. Para este sistema el espacio fase es 4-dimensional. El Hamiltoniano generalizado en (7.13) posee una simetría dihedral rotacional $\theta \rightarrow \theta + \frac{2\pi}{N}$ y para un valor par de N está presente una simetría de reflexión $r \rightarrow -r$. En cuestiones de cantidades conservadas, para valores de $N = 1, 2$, el sistema es superintegrable e integrable, respectivamente; mientras que en $N > 2$ se presentan trayectorias caóticas. Comparando el caso particular $N = 3$ con $m = \omega = 1$ la generalización (7.14) coincide con (7.1). En coordenadas cartesianas $x = r \cos \theta$, $y = r \sin \theta$, (7.13) se escribe como

$$\begin{aligned} \mathcal{H}_N &= \frac{1}{2m} (p_x^2 + p_y^2) + V_N(x, y) \\ &= \frac{1}{2m} (p_x^2 + p_y^2) + \frac{m\omega^2}{2} (x^2 + y^2) + P_N(x, y), \end{aligned} \quad (7.15)$$

con p_x , p_y como los momentos canónicos correspondientes y $P_N(x, y)$ es un polinomio homogéneo de orden N con $N > 1$. Para valores bajos de N el potencial V_N (7.14) en

coordenadas x, y viene dado por

$$\begin{aligned}
 V_1 &= \frac{m\omega^2}{2} (x^2 + y^2) + y , \\
 V_2 &= \frac{m\omega^2}{2} (x^2 + y^2) + xy , \\
 V_3 &= \frac{m\omega^2}{2} (x^2 + y^2) + yx^2 - \frac{1}{3}y^3 , \\
 V_4 &= \frac{m\omega^2}{2} (x^2 + y^2) + x^3y - y^3x , \\
 V_5 &= \frac{m\omega^2}{2} (x^2 + y^2) + yx^4 - 2x^2y^3 + \frac{1}{5}y^5 , \\
 V_6 &= \frac{m\omega^2}{2} (x^2 + y^2) + yx^5 + xy^5 - \frac{10}{3}y^3x^3 .
 \end{aligned} \tag{7.16}$$

Para cualquier valor de $N = 1, 2, 3, \dots$ la evolución temporal de las variables x, y está definida por las ecuaciones

$$\dot{x} = \frac{\partial \mathcal{H}_N}{\partial p_x} = \frac{1}{m}p_x \quad ; \quad \dot{y} = \frac{\partial \mathcal{H}_N}{\partial p_y} = \frac{1}{m}p_y , \tag{7.17}$$

que son expresiones lineales, mientras que la evolución de los momentos p_x, p_y , es diferente para cada valor de N y se obtiene con las ecuaciones de Hamilton

$$\dot{p}_x = -\frac{\partial \mathcal{H}_N}{\partial x} \quad ; \quad \dot{p}_y = -\frac{\partial \mathcal{H}_N}{\partial y} . \tag{7.18}$$

A partir de $N = 3$, en estas ecuaciones (7.18) para los momentos se obtienen expresiones no lineales, mientras que con $N = 1$, al realizar el cambio canónico de variables $x \rightarrow x, y \rightarrow y - \frac{1}{m\omega^2}$, y sin cambiar p_x, p_y , se reproduce el oscilador armónico isotrópico bidimensional. El sistema $\mathcal{H}_{N=1}$ y su contra parte cuántica son maximalmente superintegrables, se recomienda al lector revisar [28], [29]. En particular se tienen las siguientes ecuaciones dinámicas no lineales para algunos casos de N :

$N = 3$

$$\begin{aligned}
 \dot{p}_x &= -(m\omega^2 x + 2yx) , \\
 \dot{p}_y &= -(m\omega^2 y + x^2 - y^2) .
 \end{aligned} \tag{7.19}$$

$N = 4$

$$\begin{aligned}\dot{p}_x &= -(m\omega^2 x + 3yx^2 - y^3) , \\ \dot{p}_y &= -(m\omega^2 y + x^3 - 3y^2 x) .\end{aligned}\tag{7.20}$$

$N = 5$

$$\begin{aligned}\dot{p}_x &= -(m\omega^2 x + 4yx^3 - 4xy^3) , \\ \dot{p}_y &= -(m\omega^2 y + x^4 - 6y^2x^2 + y^4) .\end{aligned}\tag{7.21}$$

Por simplicidad, en las secciones siguientes, al usar estas ecuaciones se consideran los valores $m = 1$ y $\omega = 1$.

Existe para cada valor de N una energía crítica (energía de escape) en términos de este mismo parámetro, la cual se representa como

$$E_c(N) \equiv \frac{N-2}{2N} ,\tag{7.22}$$

las trayectorias del sistema transitan de un movimiento ligado a uno no ligado en este valor de energía ($E = E_c$). El valor de la energía crítica es una función monótonamente creciente en función de N ; va desde $-\frac{1}{2}$ en $N = 1$, hasta $\frac{1}{2}$ para $N \rightarrow \infty$. En la Figura 7.4 se muestran las curvas de nivel en el plano (x, y) del potencial $V_N(x, y)$ (7.16) para los casos $N = 3, 4, 5, 6$.

7.2.1. Sección de Poincaré y exponentes de Lyapunov

Para el modelo generalizado de Hénon-Heiles (7.15) se construyeron las secciones de Poincaré en el plano (y, p_y) por medio de integración numérica con el método Runge-Kutta de orden 4 (Sección 4.4) de las ecuaciones (7.17), (7.19), (7.20) para diferentes condiciones iniciales aleatorias (la construcción analítica de las condiciones iniciales se muestra en el Apéndice D), considerando $m = 1$, $\omega = 1$ para los casos $N = 3$, $N = 4$ como función de la energía E . De forma complementaria para el análisis de la naturaleza de las trayectorias, con apoyo del Dr. Mario Alan Quiróz (CFATA) quien calculó los máximos exponentes promedio de Lyapunov, se cuantifica la divergencia exponencial promedio de trayectorias que comenzaron infinitesimalmente cerca. Esto proporciona conocimientos valiosos sobre la sensibilidad del sistema a las condiciones

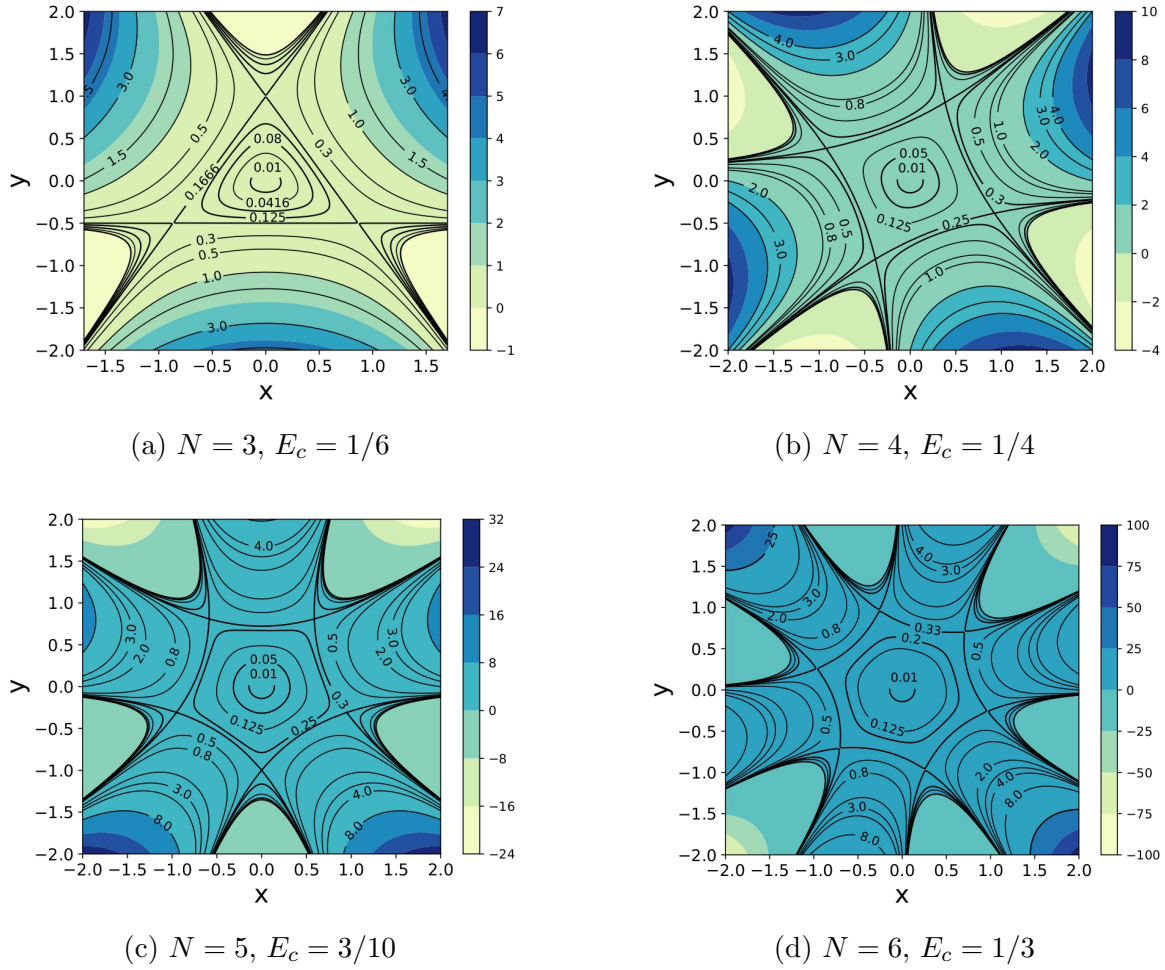


Figura 7.4: Curvas de nivel en el plano (x, y) , del potencial $V_N(x, y)$ (7.16) como función de N junto con los correspondientes valores críticos de energía E_c .

iniciales, permitiendo así caracterizar el comportamiento caótico en sistemas dinámicos. Dichas secciones de superficie de Poincaré y los exponentes de Lyapunov se ilustran en las Figuras 7.5, 7.6, de los casos $N = 3$, $N = 4$, respectivamente, considerando tres diferentes energías: $\frac{1}{4}E_c$, $\frac{3}{4}E_c$ y $\frac{99}{100}E_c$. En específico para construir las superficies de Poincaré se utilizaron 70 condiciones iniciales aleatorias con un tiempo de simulación de 7000 segundos usando Python; los exponentes de Lyapunov se construyeron integrando numéricamente 30000 condiciones iniciales en un tiempo de 7000 segundos, realizados en Mathematica 12.1.

En la Fig. 7.5 (a), (d), se observa que las secciones de Poincaré exhiben una dinámi-

ca regular cuando la energía es baja ($E = 1/4E_c$), lo cual se corrobora mediante la proximidad de los exponentes de Lyapunov a cero. Sin embargo, incrementando la energía, en el caso $E = 3/4E_c$ (Fig. 7.5 (b), (e)), las secciones de Poincaré muestran regiones con puntos dispersos, por lo que se pierde la regularidad, para este incremento de energía los exponentes de Lyapunov muestran valores diferentes de cero, lo que revela la presencia de caos en el sistema. Es relevante resaltar que ciertas regiones donde se observa un comportamiento periódico son fácilmente identificables gracias a la presencia de zonas más oscuras en el gráfico. Para una energía demasiado cercana a la crítica E_c la presencia de caos es dominante, como se muestra en la Fig. 7.5(c), (f), despreciando esto ciertas islas de orden persisten, revelando la coexistencia entre regularidad y caos. El mismo comportamiento sucede al considerar los mismos incrementos de energía en el caso $N = 4$, Fig. 7.6, donde la regularidad comienza a desaparecer y el caos se hace presente.

7.2.2. Líneas de simetría

Adicional a las secciones de superficie de Poincaré que se obtuvieron, se construyen algunas líneas de simetría (Sección 4.3.2), para obtener mediante sus intersecciones, puntos en la sección de superficie que corresponden a órbitas periódicas, los cuales son utilizados como condiciones iniciales para estudiar este sistema con el algoritmo de aprendizaje automático. Este sistema generalizado de Hénon-Heiles para los casos $N = 3$ - $N = 5$, (7.19)-(7.21), exhiben la propiedad de reversibilidad. En concreto, las ecuaciones de Hamilton son invariantes, cuando se invierte el sentido del tiempo y el momento, esto definido de la forma:

$$(\mathbf{q}, \mathbf{p}, t) \rightarrow (\mathbf{q}, -\mathbf{p}, -t) ; \quad (7.23)$$

asimismo, para valores impares del parámetro N , se posee una simetría de coordenadas e inversión temporal:

$$(\mathbf{q}, \mathbf{p}, t) \rightarrow (-\mathbf{q}, \mathbf{p}, -t) . \quad (7.24)$$

Para estos sistemas reversibles, como se describe en la Sección 4.3.2 y en [6], la sección de superficie de Poincaré puede ser expresada por la composición de dos involuciones,

mientras que los puntos estacionarios bajo la involución I_i se definen como las líneas de simetría (4.33):

$$\Gamma_i : \{\mathbf{x} \in S | I_i \mathbf{x} = \mathbf{x}\} \quad , \quad \text{para toda } i \in \mathbb{Z} . \quad (7.25)$$

En particular las líneas fundamentales Γ_0 y Γ_1 , conducen a la propiedad (4.34):

$$T^i \Gamma_j = \Gamma_{2i+j} , \quad (7.26)$$

que implica la generación de otras líneas con índices pares: $\Gamma_{2i} = T^i \Gamma_0$ y con índices impares: $\Gamma_{2i+1} = T^i \Gamma_1$, respectivamente, como se menciona en la Sección 4.3.2. En el caso de la línea Γ_2 , esta se obtiene mediante,

$$\Gamma_2 = T^1 \Gamma_0 \quad , \text{ con } i = 1 . \quad (7.27)$$

Cabe resaltar que las órbitas periódicas correspondientes a intersecciones entre líneas de simetría, $\Gamma_i \cap \Gamma_j$ tienen periodo divisor del entero $|j - i|$. En el sistema generalizado de Hénon-Heiles, para el caso $N = 3$, la involución I_0 es usada en el plano Σ (sección de Poincaré) que corresponde a la simetría (7.23) y junto con los puntos fijos en Γ_0 se tiene:

$$I_0(x, y, p_x, p_y, t) = (x, y, -p_x, -p_y, t) \quad , \quad \Gamma_0 = \{(y, p_y) | p_y = 0\} , \quad (7.28)$$

mientras que en el caso $N = 4$ para la simetría de coordenadas e inversión del tiempo (7.24),

$$I_0(x, y, p_x, p_y, t) = (-x, -y, p_x, p_y, -t) \quad , \quad \Gamma_0 = \{(y, p_y) | y = 0\} . \quad (7.29)$$

En concreto, para $N = 3$ y la simetría (7.28), las órbitas periódicas encontradas corresponden a la intersección de los puntos sobre la línea horizontal Γ_0 (línea negra) con Γ_2 (línea azul), Fig. 7.5(a)-(c), para los diferentes niveles de energía; las coordenadas en el plano p_y vs y de estas intersecciones se muestran en el Cuadro 7.1. Mientras que en el caso $N = 4$, con la simetría (7.29), los puntos que se obtienen de la intersección de la línea vertical Γ_0 (línea negra) con Γ_2 (línea azul), Fig. 7.6 (a)-(c), se presentan en el Cuadro 7.2.

E	(x, y, p_x, p_y)
$1/4E_c$	$(0, -0,2134150807894881, 0,176561230871, 0)$
$3/4E_c$	$(0, 0,3038006035414, 0,41999762532241397, 0)$
$99/100E_c$	$(0, 0,5473794640724625, 0,37821665749647776, 0)$

Cuadro 7.1: Coordenadas correspondientes a órbitas periódicas obtenidas mediante la intersección de las líneas Γ_0 y Γ_2 para diferentes valores de energía en el caso $N = 3$ del sistema generalizado de Hénon-Heiles, ilustradas en Fig. 7.5 (a)-(c).

E	(x, y, p_x, p_y)
$1/4E_c$	$(0, 0,13523834118944772, 0, 0,3266658706879654)$
$3/4E_c$	$(0, -0,23430033115993099, 0, 0,5657767711901459)$
$99/100E_c$	$(0, -0,6529459864224533, 0, 0,27140659316753796)$

Cuadro 7.2: Coordenadas correspondientes a órbitas periódicas obtenidas mediante la intersección de las líneas Γ_0 y Γ_2 para diferentes valores de energía en el caso $N = 4$ del sistema generalizado de Hénon-Heiles, ilustradas en Fig. 7.6 (a)-(c).

7.3. Reconstrucción de las ecuaciones de Hamilton con SINDy

Se considera como un modelo de prueba la generalización del sistema de Hénon-Heiles (7.15) con el que se analiza la precisión del algoritmo SINDy (Capítulo 6), bajo la presencia de caos para los diferentes niveles de energía $E = \frac{1}{4}E_c, \frac{3}{4}E_c, \frac{99}{100}E_c$, en los casos $N = 3, N = 4$ para los cuales se obtuvieron las secciones de superficie de Poincaré y exponentes de Lyapunov. La elección de condiciones iniciales caóticas y cuasiperiódicas, con las que se construyen las series temporales mediante integración numérica (RK-45[12]) de las ecuaciones (7.19), (7.20) en un intervalo de tiempo $[t_i = 0, t_f = 300]$, que alimentan al algoritmo SINDy es trivial analizando las secciones de superficie y los exponentes de Lyapunov en la Fig. 7.5, 7.6; en el caso de las condiciones iniciales de órbitas periódicas, se consideran los puntos obtenidos de forma aproximada mediante intersecciones de las líneas de simetría Γ_0, Γ_2 , Cuadros 7.1 y 7.2. En particular para la energía $\frac{1}{4}E_c$ solo se consideran condiciones iniciales de orbitas periódicas y cuasiperiódicas ya que no hay una presencia significativa de caos, como se puede observar en las Figuras 7.5, 7.6 (d).

En las Figuras 7.5, 7.6 se marcan los puntos que corresponden a las condiciones ini-

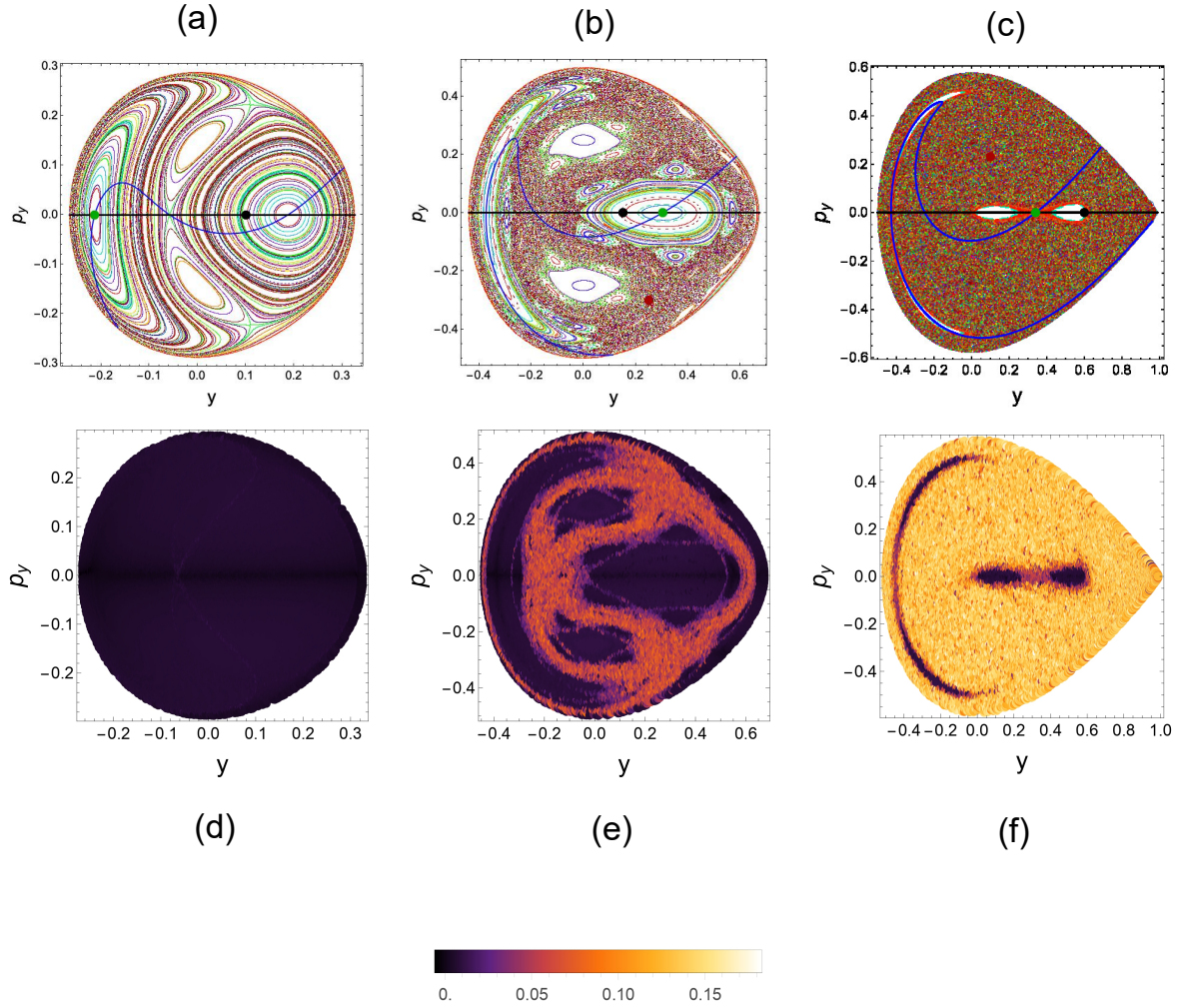


Figura 7.5: Caso $N = 3$: (a)-(c) secciones de Poincaré orientadas y líneas de simetría Γ_0 y Γ_2 (7.25), y (d)-(f) los exponentes de Lyapunov en el plano (y, p_y) , para el Hamiltoniano $\mathcal{H}_{N=3}$ (7.15) con $m = 1$, $\omega = 1$ en diferentes valores de energía $E = \frac{1}{4}E_c$, $\frac{3}{4}E_c$, $\frac{99}{100}E_c$. En este caso, la energía crítica es $E_c = \frac{1}{6}$. Los puntos verdes, negros y rojos marcados en (a)-(c) indican las condiciones iniciales para trayectorias periódicas, cuasi periódicas y caóticas, respectivamente; éstas fueron usadas para construir Fig. 7.7 (a)-(c). Figura tomada de [7].

ciales seleccionadas y sus coordenadas (x, y, p_x, p_y) se muestran en los cuadros 7.3-7.5. Para $N = 3$ y $N = 4$, se obtienen las siguientes condiciones iniciales en los Cuadros 7.3-7.8.

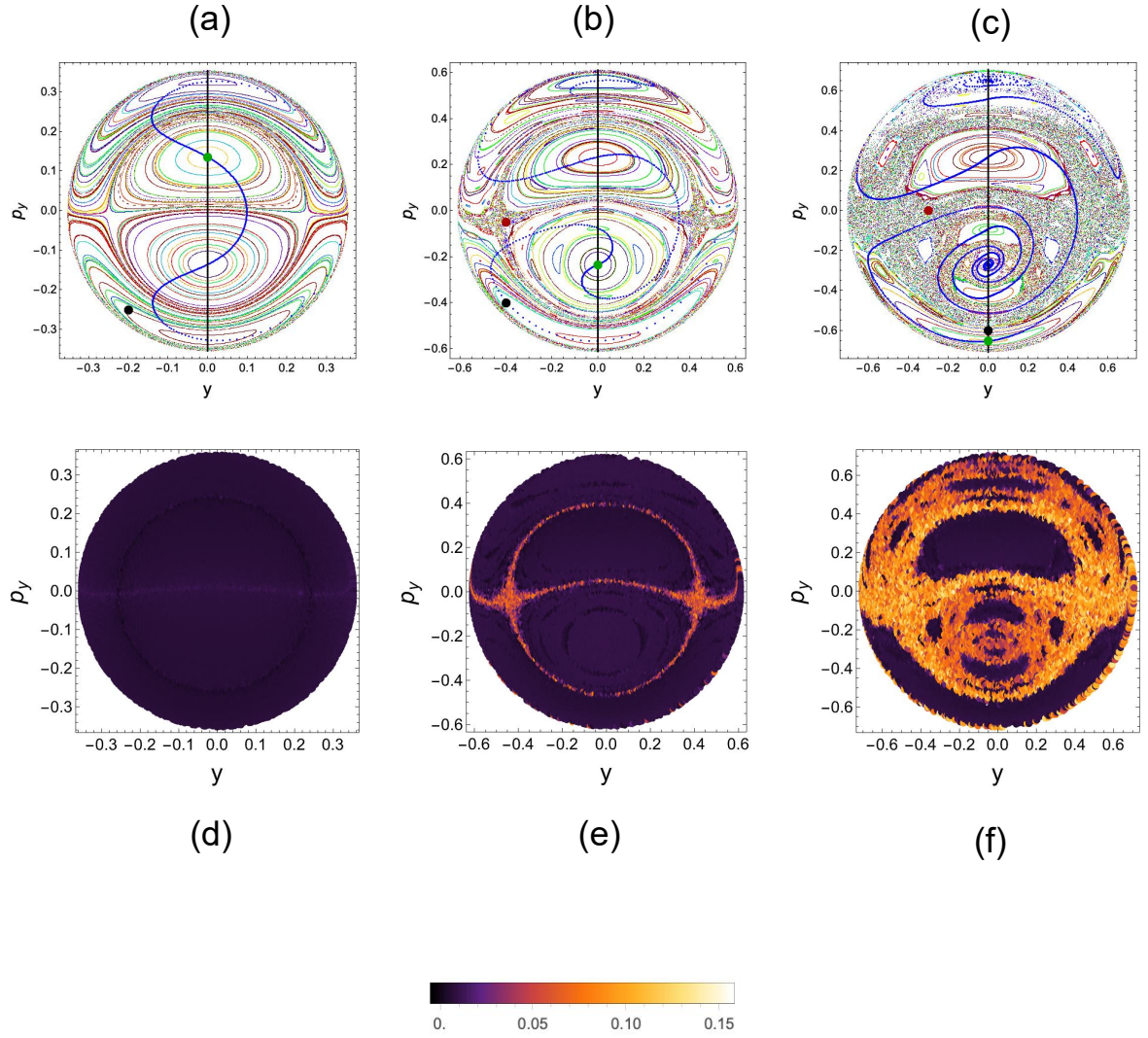


Figura 7.6: Caso $N = 4$: (a)-(c) secciones de Poincaré orientadas y líneas de simetría Γ_0 y Γ_2 (7.25), y (d)-(f) los exponentes de Lyapunov en el plano (y, p_y) , para el Hamiltoniano $\mathcal{H}_{N=3}$ (7.15) con $m = 1$, $\omega = 1$ en diferentes valores de energía $E = \frac{1}{4}E_c$, $\frac{3}{4}E_c$, $\frac{99}{100}E_c$. En este caso, la energía crítica es $E_c = \frac{1}{6}$. Los puntos verdes, negros y rojos marcados en (a)-(c) indican las condiciones iniciales para trayectorias periódicas, cuasiperiódicas y caóticas, respectivamente; éstas fueron usadas para construir Fig. 7.7 (d)-(f). Figura tomada de [7].

Para todas las condiciones iniciales de los Cuadros 7.3 - 7.8 se calcularon, por medio de integración numérica (Runge-Kutta-Fehlberg [12]) de las ecuaciones (7.19), (7.20)

7.3. RECONSTRUCCIÓN DE LAS ECUACIONES DE HAMILTON CON SINDY75

$E = 1/4E_c$	(x, y, p_x, p_y)
Periódica	$(0, -0,2134150807894881, 0,176561230871, 0)$
Cuasiperiódica	$(0, 0,1, 0,271784, 0)$

Cuadro 7.3: Condiciones iniciales para la energía $E = \frac{1}{4}E_c$ del caso $N = 3$.

$E = 3/4E_c$	(x, y, p_x, p_y)
Periódica	$(0, 0,3038006035414, 0,41999762532241397, 0)$
Cuasiperiódica	$(0, 0,15, 0,479322, 0)$
Caótica	$(0, 0,25, 0,328507, -0,3)$

Cuadro 7.4: Condiciones iniciales para la energía $E = \frac{3}{4}E_c$ del caso $N = 3$.

$E = 99/100E_c$	(x, y, p_x, p_y)
Periódica	$(0, 0,342446033, 0,492649, 0)$
Cuasiperiódica	$(0, 0,6, 0,342345, 0)$
Caótica	$(0, 0,1, 0,520545, 0,23)$

Cuadro 7.5: Condiciones iniciales para la energía $E = \frac{99}{100}E_c$ del caso $N = 3$.

$E = 1/4E_c$	(x, y, p_x, p_y)
Periódica	$(0, 0, 0,3266658706879654, 0,13523834118944772)$
Cuasiperiódica	$(0, -0,2, 0,15, -0,25)$

Cuadro 7.6: Condiciones iniciales para la energía $E = \frac{1}{4}E_c$ del caso $N = 4$.

$E = 3/4E_c$	(x, y, p_x, p_y)
Periódica	$(0, 0, 0,5657767711901459, -0,23430033115993099)$
Cuasiperiódica	$(0, -0,4, 0,234521, -0,4)$
Caótica	$(0, -0,4, 0,460977, -0,05)$

Cuadro 7.7: Condiciones iniciales para la energía $E = \frac{3}{4}E_c$ del caso $N = 4$.

$E = 99/100E_c$	(x, y, p_x, p_y)
Periódica	$(0, 0, 0,27140659316753796, -0,6529459864224533)$
Cuasiperiódica	$(0, 0, 0,374166, -0,6)$
Caótica	$(0, -0,3, 0,640312, 0)$

Cuadro 7.8: Condiciones iniciales para la energía $E = \frac{99}{100}E_c$ del caso $N = 4$.

con un paso de integración Δt característico; las series temporales correspondientes

que se pueden visualizar en el Apéndice F.

En específico, usando como condiciones iniciales las coordenadas indicadas de los Cuadros 7.3 - 7.8, se realiza una reconstrucción de las ecuaciones de Hamilton para poder examinar la precisión de SINDy en función de valores fijos para la energía y la transición de órbitas periódicas a órbitas caóticas; para cada una se ilustra su representación en los planos (x, y) , Fig. 7.7 y (y, p_y) , Fig.F.2 en el Apéndice F.

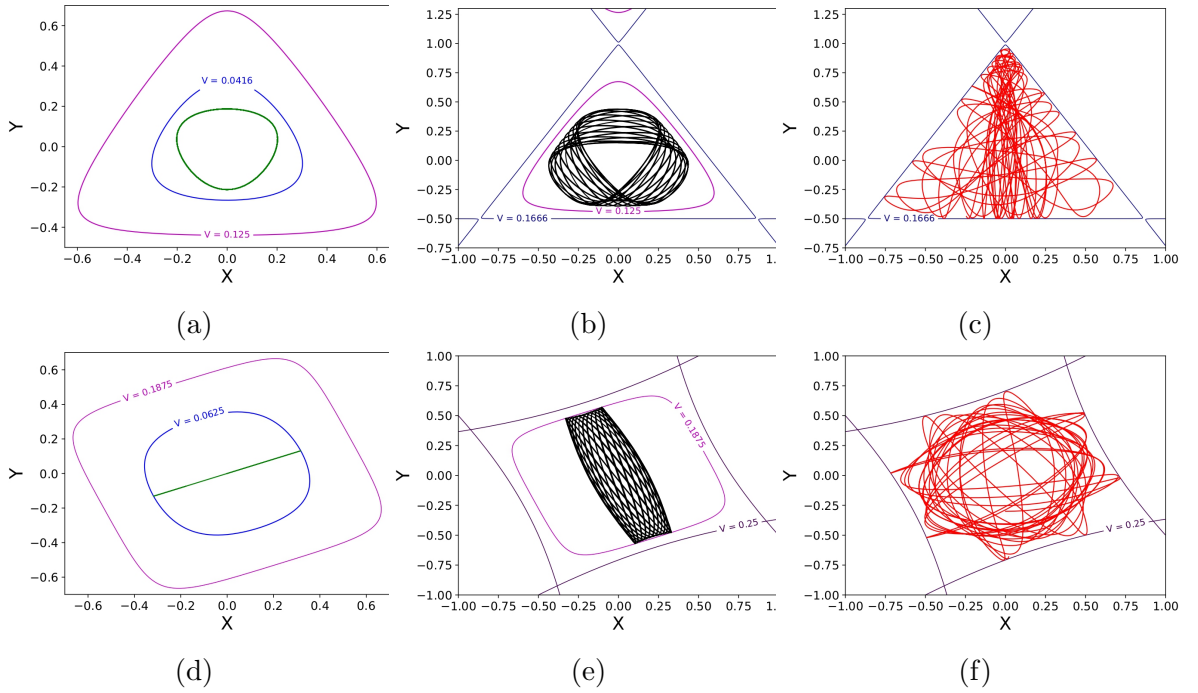


Figura 7.7: En el panel superior se muestran las curvas de nivel en el plano (x, y) para el caso de $N = 3$ y considerando el potencial V_3 (7.16) con $m\omega^2 = 1$. La energía crítica $E_c = \frac{1}{6} \approx 0,1666$ se indica así como las trayectorias: (a) periódica (línea verde) con la condición iniciales $(x_0 = 0, y_0 = -0,21341508, p_{x_0} = 0,17656123, p_{y_0} = 0)$ y energía $E = \frac{1}{4}E_c$, (b) cuasiperiódica (línea negra) con la condición inicial $(x_0 = 0, y_0 = 0,15, p_{x_0} = 0,479322, p_{y_0} = 0)$ en la energía $E = \frac{3}{4}E_c$ y por último (c) caótica (línea roja) con condición inicial $(x_0 = 0, y_0 = 0,1, p_{x_0} = 0,520545, p_{y_0} = 0,23)$ y $E = \frac{99}{100}E_c$. En el panel inferior se muestran las curvas de nivel en el plano (x, y) para el caso de $N = 4$ y el potencial V_4 (7.16) con $m\omega^2 = 1$. La energía crítica $E_{\text{crit}} = \frac{1}{4} = 0,25$ se indica así como las trayectorias: (d) periódica (línea verde) con la condición $(x_0 = 0, y_0 = 0, p_{x_0} = 0,32666587, p_{y_0} = 0,13523834)$ y energía $E = \frac{1}{4}E_c$, (e) cuasiperiódica (línea negra) de la condición inicial $(x_0 = 0, y_0 = -0,4, p_{x_0} = 0,234521, p_{y_0} = -0,4)$ y energía $E = \frac{3}{4}E_c$, por último (f) caótica (línea roja) con condición inicial $(x_0 = 0, y_0 = -0,3, p_{x_0} = 0,640312, p_{y_0} = 0)$ y energía $E = \frac{99}{100}E_c$. Figura tomada de [7].

El algoritmo se nutre de las series temporales de las condiciones iniciales mencionadas y además requiere una base de funciones candidatas para la derivación del modelo. Dado que se conocen las ecuaciones exactas (7.19), (7.20), para cada ecuación $\dot{x}, \dot{p}_x, \dot{y}, \dot{p}_y$ la opción es usar funciones polinomiales de grado $K > N - 1$ en las variables de estado x, y, p_x, p_y para las cuales los C -coeficientes de cada ecuación son las incógnitas. Por ejemplo, el ansatz polinomial más general de orden K es de la forma

$$\dot{x} = \sum_{0 \leq n_1 + n_2 + n_3 + n_4 = K} C_x(n_1, n_2, n_3, n_4) x^{n_1} y^{n_2} p_x^{n_3} p_y^{n_4}, \quad (7.30)$$

donde las $C_\tau(\sigma)$ constantes son parámetros reales. Expresiones similares se pueden escribir para las otras ecuaciones de estado.

Siguiendo el diagrama de la Figura 6.3, considerando que las series temporales no están contaminados (no tienen ruido), se hace uso de una técnica de regularización arbitraria (optimizador), en este caso se usa STLQ descrito en la sección 6.1, propuesto en [16].

Por lo tanto, para cada condición inicial y su respectiva serie temporal, junto con un valor del hiperparámetro λ para el optimizador, que se obtiene utilizando la técnica *frente de Pareto* (procedimiento ilustrado en la Fig. 7.8), se calculan modelos parsimonios con SINDy para diferentes valores del paso de integración Δt . En la Fig. 7.9 se muestra un ejemplo para la elección del hiperparámetro λ con el cual se obtienen los modelos óptimos de cada valor Δt , en estos se revisa la interpolación del modelo obtenido ante los datos de entrenamiento y la extrapolación a datos de prueba en función del ϵ_{MSE} .

Esencialmente, se muestra en los Cuadros 7.9- 7.11 la reconstrucción de las ecuaciones de Hamilton para algunos valores del paso de integración Δt de las trayectorias periódicas, cuasiperiódica y caótica del caso $N = 3$ seleccionadas de 7.3-7.5. Lo mismo para el caso $N = 4$ en los Cuadros 7.12-7.14 de las trayectorias seleccionadas de 7.6-7.8.

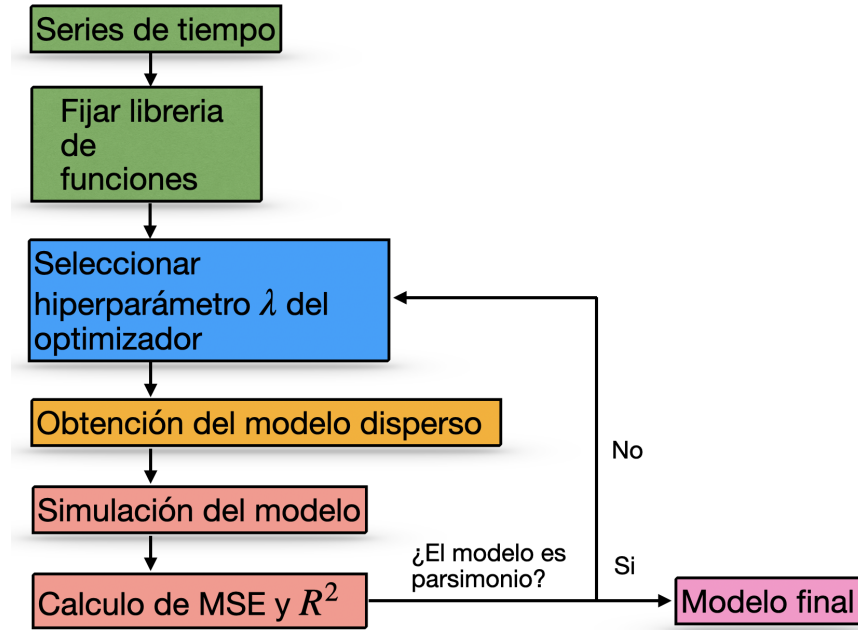


Figura 7.8: Procedimiento para derivar un modelo parsimonioso usando SINDy a través del frente de Pareto, lo cual implica la selección del valor óptimo de λ .

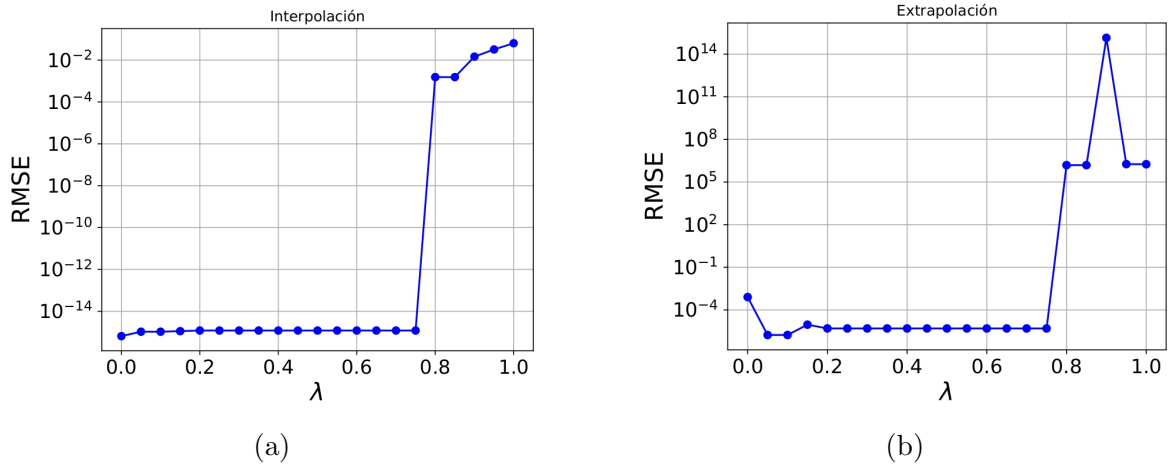


Figura 7.9: Para la órbita periódica con energía $E = \frac{1}{4}E_c$ del caso $N = 3$ se usa la técnica frente de Pareto para 11 valores de $\lambda \in [0, 1]$. En específico para este ejemplo el modelo extrapola con el menor error cuadrático medio (ϵ_{MSE}) para valores de $\lambda = 0,5$ y $\lambda = 0,1$.

dt	Modelo SINDy	ϵ_{MSE}	R^2
0.0001	$\dot{x} = 0,999999992p_x$ $\dot{y} = 0,999999997p_y$ $\dot{p}_x = -1,00000461x - 2,00000927xy$ $\dot{p}_y = -1,00000269y - 1,00000446x^2 + 1,00000535y^2$	$4,32551 \times 10^{-11}$	0.9999999992
0.001	$\dot{x} = 0,999999794p_x$ $\dot{y} = 1,00000103p_y$ $\dot{p}_x = -1,00000184x - 1,99999858xy$ $\dot{p}_y = -0,999988327y - 1,0000237x^2 + 0,999956598y^2$	$3,89755 \times 10^{-10}$	0.9999999935
0.01	$\dot{x} = 0,999982640p_x$ $\dot{y} = 9,99984627p_y$ $\dot{p}_x = -0,999977583x - 1,99985508xy$ $\dot{p}_y = -1,00000134y - 0,999888083x^2 + 1,00002048y^2$	$1,34483 \times 10^{-6}$	0.9999778238
0.017 dt_c	$\dot{x} = 9,99945532p_x$ $\dot{y} = 9,99952969p_y$ $\dot{p}_x = -0,999937324x - 1,99958899xy$ $\dot{p}_y = -1,00000125y - 0,999694563x^2$ $+1,00001704y^2 + 0,001y^3$	$1,19281 \times 10^{-5}$	0.9998033077
0.09	$\dot{x} = 0,001y + 0,998p_x - 0,007x^2$ $-0,016xp_y - 0,006y^2 + 0,013yp_x$ $-0,008p_x^2 - 0,006p_y^2 - 0,002x^2p_x$ $\dot{y} = 0,999p_y - 0,003xy - 0,002xp_x$ $+0,002yp_y - 0,001p_xp_y$ $\dot{p}_x = -0,999x - 1,989xy - 0,001xp_x$ $+0,002yp_y - 0,005p_xp_y + 0,003x^3$ $\dot{p}_y = -1,011y + 0,004p_x - 0,969x^2$ $+0,043xp_y + 1,029y^2 - 0,058yp_x$ $+0,026p_x^2 + 0,021p_y^2 + 0,123y^3$ $-0,050y^2p_x + 0,058yp_y^2$	0.00844	0.8607

Cuadro 7.9: Reconstrucción de las ecuaciones de Hamilton en el caso $N = 3$, de la órbita periódica marcada en rojo del Cuadro 7.3 con energía $E = \frac{1}{4}E_c$, para diferentes valores del intervalo en el tiempo Δt , usando el hiperparámetro $\lambda = 0,05$ obtenido mediante frente de Pareto. Existe un dt_c aproximado a partir del cual, términos extras a las ecuaciones de Hamilton se hacen presentes.

7.3.1. Resultados

Para determinar la precisión del algoritmo, se introduce el parámetro de error $\Delta C_\tau(\sigma)$, como

$$\Delta C_\tau(\sigma) = \frac{C_\tau^{\text{exact}}(\sigma) - C_\tau^{\text{SINDy}}(\sigma)}{C_\tau^{\text{exact}}(\sigma)}, \quad (7.31)$$

dt	Modelo SINDy	ϵ_{MSE}	R^2
0.0001	$\dot{x} = 0,99999999p_x$ $\dot{y} = 0,99999999p_y$ $\dot{p}_x = -0,99999997x - 1,99999971xy$ $\dot{p}_y = -0,99999998y - 0,99999987x^2$ $+0,99999987y^2$	$7,36312 \times 10^{-11}$	0.9999999993
0.001	$\dot{x} = 0,9999998p_x$ $\dot{y} = 0,99999981p_y$ $\dot{p}_x = -0,99999978x - 1,99999844xy$ $\dot{p}_y = -0,99999979y - 0,99999923x^2$ $+0,99999922y^2$	$4,30505 \times 10^{-10}$	0.9999999881
0.02726 dt_c	$\dot{x} = 0,999827406p_x$ $\dot{y} = 0,999826303p_y$ $\dot{p}_x = -0,999819439x - 1,99914793xy$ $-0,001p_xp_y$ $\dot{p}_y = -0,999834687y - 0,999387966x^2$ $+0,999384886y^2$	$7,06937 \times 10^{-5}$	0.998862871
0.1	$\dot{x} = 0,998p_x$ $\dot{y} = 0,998p_y$ $\dot{p}_x = -0,998x + -1,990xy$ $-0,007p_xp_y - 0,003x^2p_y$ $-0,002xp_x^2 - 0,004xp_y^2$ $-0,001yp_xp_y$ $\dot{p}_y = -0,999834687y - 0,999387966x^2$ $+0,999384886y^2$	0,008429	0.864409236

Cuadro 7.10: Reconstrucción de las ecuaciones de Hamilton en el caso $N = 3$, de la órbita cuasiperiódica marcada en rojo del Cuadro 7.4 con energía $E = \frac{3}{4}E_c$, para diferentes valores del intervalo en el tiempo Δt , usando el hiperparámetro $\lambda = 0,05$ obtenido mediante frente de Pareto. Existe un dt_c aproximado a partir del cual, términos extras a las ecuaciones de Hamilton se hacen presentes.

donde $C_\tau^{\text{exact}}(\sigma)$ representa el coeficiente correspondiente al término σ en la ecuación de estado τ . Estas constantes $C_\tau(\sigma)$ dependen del paso de integración Δt correspondiente al intervalo de tiempo usado para la integración numérica de las ecuaciones de movimiento, el mismo que está asociado a la diferencia entre cada dato de la serie temporal. Para aclarar la notación usada, se examina como ejemplo el caso de $N = 3$; la ecuación exacta que corresponde a la derivada de p_x respecto al tiempo es $\dot{p}_x = -x - 2yx$, así

7.3. RECONSTRUCCIÓN DE LAS ECUACIONES DE HAMILTON CON SINDY81

dt	Modelo SINDy	ϵ_{MSE}	R^2
0.0001	$\dot{x} = 0,99999999p_x$ $\dot{y} = 0,99999999p_y$ $\dot{p}_x = -0,99999997x - 1,9999999xy$ $\dot{p}_y = -0,99999997y - 0,99999996x^2$ $+0,99999995y^2$	0,133605	-0.456589
0.001	$\dot{x} = 0,99999981p_x$ $\dot{y} = 0,99999985p_y$ $\dot{p}_x = -0,99999969x - 1,99999899xy$ $\dot{p}_y = -0,99999972y - 0,99999955x^2$ $+0,99999962y^2$	0,145638	-0.58473346
0.01	$\dot{x} = 0,99998248p_x$ $\dot{y} = 0,99998462p_y$ $\dot{p}_x = -0,99997332x - 1,99991132xy$ $\dot{p}_y = -0,99997428y - 0,99995926x^2$ $+0,99996411y^2$	0,1486236	-0.59618081
0.1	$\dot{x} = 0,99805072p_x$ $\dot{y} = 0,99851295p_y$ $\dot{p}_x = -0,99730997x - 1,99025069xy$ $\dot{p}_y = -0,99752516y - 0,99554181x^2$ $+0,99649843y^2$	0,17172956	-1.17548968

Cuadro 7.11: Reconstrucción de las ecuaciones de Hamilton en el caso $N = 3$, de la órbita caótica marcada en rojo del Cuadro 7.5 con energía $E = \frac{99}{100}E_c$, para diferentes valores del intervalo en el tiempo Δt , usando el hiperparámetro $\lambda = 0,05$ obtenido mediante frente de Pareto. En este caso no existe valor límite dt_c .

que $\tau = \dot{p}_x, \sigma = x, xy$ y los coeficientes diferentes de cero son,

$$C_{\dot{p}_x}^{exact}(x) = -1 \quad , \quad C_{\dot{p}_x}^{exact}(xy) = -2 \quad . \quad (7.32)$$

De forma similar, se pueden escribir con esta notación los coeficientes que corresponden a las ecuaciones obtenidas por SINDy, $C_{\tau}^{SINDy}(\sigma)$. Naturalmente, y como se puede observar en el Cuadro 7.9 del caso $N = 3$, al incrementar el valor Δt , el modelo comienza a perder precisión mientras aumenta el error cuadrático medio y el coeficiente de correlación disminuye debido a que SINDy obtiene otros valores diferentes de los exactos para los C -coeficientes.

Las ecuaciones reconstruidas para valores suficientemente pequeños de Δt contienen

dt	Modelo SINDy	ϵ_{MSE}	R^2
0.00002	$\dot{x} = 1,000003p_x$ $\dot{y} = 0,999997p_y$ $\dot{p}_x = 0,003218y - 1,001318x - 0,757876y^2x$ $+2,818599yp_y p_x - 1,889349yx^2 - 1,197743yp_x^2$ $-1,168654p_y xp_x - 0,259088x^3 + 0,496456xp_x^2$ $\dot{p}_y = -1,000010y + 0,000010x + 2,485279yx^2$ $-0,000423p_y xp_x - 1,514775x^3 + 0,000116xp_x^2$	$1,3592 \times 10^{-13}$	0.9999999999
0.001	$\dot{x} = 1,0000032p_x$ $\dot{y} = 0,999996366p_y$ $\dot{p}_x = 0,003468y - 1,001423x - 2,782016y^2x$ $+1,857990yp_y p_x - 0,215099yx^2 - 0,802159yp_x^2$ $-2,507656p_y^2x + 1,306615p_y xp_x - 0,605288x^3$ $-0,097624xp_x^2$ $\dot{p}_y = -0,998476y - 0,000626x + 2,470112yx^2$ $-0,014289yp_x^2 - 0,000476p_y xp_x - 1,508490x^3$ $+0,006059xp_x^2$	$5,00092 \times 10^{-12}$	0.9999999998
0.01	$\dot{x} = 1,000019p_x$ $\dot{y} = 0,999949p_y$ $\dot{p}_x = -0,001202y - 0,999471x + 2,125546y^2x$ $+0,646269yp_y p_x - 4,234009yx^2 - 0,256645yp_x^2$ $+1,639951p_y^2x - 1,626526p_y xp_x + 0,217464x^3$ $+0,387521xp_x^2$ $\dot{p}_y = -1,000389y + 0,000174x + 2,489167yx^2$ $+0,003841yp_x^2 + 0,000140p_y xp_x - 1,516353x^3$ $-0,001757xp_x^2$	$3,65638 \times 10^{-8}$	0.9999988039

Cuadro 7.12: Reconstrucción de las ecuaciones de Hamilton en el caso $N = 4$, de la órbita periódica marcada en rojo del Cuadro 7.6 con energía $E = \frac{1}{4}E_c$, para diferentes valores del intervalo en el tiempo Δt , usando el hiperparámetro $\lambda = 0,05$ obtenido mediante frente de Pareto. En este caso no existe un valor dt_c , para valores infinitesimales del paso de integración se presentan términos extras para las ecuaciones de Hamilton.

los σ términos originales, pero mientras se incrementa cada vez más el intervalo de integración, términos extra se hacen presentes en las ecuaciones, por lo que se define un valor critico dt_c a partir del cual esto sucede. En el caso $N = 4$, la aparición de términos extra es notable para cualquier valor que se elija del paso de integración, lo curioso es que a pesar de no obtener las ecuaciones exactas, las herramientas cuantitativas para evaluar el modelo (ϵ_{MSE} , R^2) indican la posibilidad de precisión en este caso; este punto

7.3. RECONSTRUCCIÓN DE LAS ECUACIONES DE HAMILTON CON SINDY83

dt	Modelo SINDy	ϵ_{MSE}	R^2
0.0001	$\dot{x} = 1p_x$ $\dot{y} = 1p_y$ $\dot{p}_x = -1,00000001x + 1,00000026y^3 - 3,00000051yx^2$ $\dot{p}_y = -0,99999998y + 3,0000003y^2x - 1,00000008x^3$	$2,16383 \times 10^{-7}$	0.99999725
0.001	$\dot{x} = 0,99999984p_x$ $\dot{y} = 0,99999981p_y$ $\dot{p}_x = -0,99999979x + 0,99999817y^3 - 2,99999633yx^2$ $\dot{p}_y = -0,99999991y + 2,99999628y^2x - 0,99999912x^3$	$9,754511 \times 10^{-6}$	0.9998765
0.0136 dt_c	$\dot{x} = 1,000221px - 0,000819y^2px$ $+0,000223yxp_x - 0,000614p_y^2px$ $+0,000526p_y p_x^2$ $\dot{y} = 0,999963p_y$ $\dot{p}_x = 0,00013y - 0,99996x$ $+0,99933y^3 - 0,00011yp_y^2$ $-2,99977yx^2 - 0,00041yp_x^2$ $\dot{p}_y = -0,999959y + -0,000041x$ $+2,999773y^2x + 0,000506yp_y p_x$ $+0,000100yp_x^2 + 0,000355p_y^2x$ $+0,000276p_y x p_x + -0,999801x^3$ $+0,000234xp_x^2$	$1,0550 \times 10^{-4}$	0.99861657

Cuadro 7.13: Reconstrucción de las ecuaciones de Hamilton en el caso $N = 4$, de la órbita cuasiperiódica marcada en rojo del Cuadro 7.7 con energía $E = \frac{3}{4}E_c$, para diferentes valores del intervalo en el tiempo Δt , usando el hiperparámetro $\lambda = 0,05$ obtenido mediante frente de Pareto. Existe un dt_c aproximado a partir del cual términos extras a las ecuaciones exactas aparecen.

será discutido a detalle más adelante.

Resultados concretos considerando la reconstrucción con el hiperparámetro óptimo $\lambda = 0,05$ (obtenido mediante frente de Pareto para cada Δt) en ambos casos $N = 3$, $N = 4$, para todo tipo de órbitas, se muestra el error relativo $\Delta C_\tau(\sigma)$ en función de Δt para las trayectorias seleccionadas en las Figs. F.3, F.5 y Figs. F.6, F.7 del Apéndice F. En general la mayoría de las curvas $\Delta C_\tau(\sigma)$ vs Δt son funciones continuas positivas que aumentan gradualmente y disminuyen a medida que $dt \rightarrow 0$.

Analizando las ecuaciones que se obtienen de las órbitas periódicas y cuasiperiódicas para diferentes valores en Δt , del caso $N = 3$, se observa en los valores de R^2 que los modelos pierden precisión significativamente a partir del valor dt_c ; mientras que en

dt	Modelo SINDy	ϵ_{MSE}	R^2
0.0001	$\dot{x} = 1p_x$ $\dot{y} = 0,99999999p_y$ $\dot{p}_x = -1,00000001x + 0,99999996y^3$ $\quad -3,00000008yx^2$ $\dot{p}_y = -1,00000001y + 2,99999966y^2x$ $\quad -0,99999993x^3$	$1,70048007 \times 10^{-9}$	0.999999980
0.001	$\dot{x} = 0,9999998p_x$ $\dot{y} = 0,99999979p_y$ $\dot{p}_x = -0,99999964x + 0,9999994y^3$ $\quad -2,99999703yx^2$ $\dot{p}_y = -1,00000006y + 2,99999453y^2x$ $\quad -0,99999553x^3$	$7,64947977 \times 10^{-10}$	0.999999988
0.0136 dt_c	$\dot{x} = 1,00017p_x - 0,00046y^2p_x$ $\quad -0,00039p_y^2p_x - 0,00019x^2p_x$ $\quad -0,00035p_x^3$ $\dot{y} = 0,99998p_y$ $\dot{p}_x = -1,00004x + 0,99992y^3$ $\quad -0,00041yp_y p_x - 2,99991yx^2$ $\quad -0,00044yp_x^2 - 0,00049p_y x p_x$ $\quad +0,00021x^3$ $\dot{p}_y = -0,99997y - 0,00002xx$ $\quad +2,99985y^2x + 0,00025yp_y p_x$ $\quad +0,00017p_y^2x + 0,00012p_y x p_x$ $\quad -0,99990x^3 + 0,00005xp_x^2$	$2,149455341 \times 10^{-6}$	0.9999832

Cuadro 7.14: Reconstrucción de las ecuaciones de Hamilton en el caso $N = 4$, de la órbita caótica marcada en rojo del Cuadro 7.8 con energía $E = \frac{99}{100}E_c$, para diferentes valores del intervalo en el tiempo Δt , usando el hiperparámetro $\lambda = 0,05$ obtenido mediante frente de Pareto. Existe un dt_c aproximado a partir del cual, términos extras a las ecuaciones de Hamilton se hacen presentes.

la trayectoria caótica, a pesar de recuperar las ecuaciones de Hamilton correctas, sin presencia del dt_c , al realizar el proceso de validación del modelo este no reproduce correctamente toda la trayectoria (Fig. 7.10 (c)). En el caso $N = 4$ se enfatiza que para la trayectoria periódica el límite del paso de integración es cero dado que para cualquier valor, incluso infinitesimal, aparecen términos extras en las ecuaciones reconstruidas, curiosamente las medidas cuantitativas indican que se tienen expresiones que cumplen con valores esperados en ϵ_{MSE} y R^2 . Finalmente para la trayectoria cuasiperiódica y

7.3. RECONSTRUCCIÓN DE LAS ECUACIONES DE HAMILTON CON SINDY85

caótica se tiene el mismo comportamiento que en $N = 3$ al incrementar Δt .

Por lo tanto, independientemente de la naturaleza del movimiento (ya sea periódico o caótico), el algoritmo SINDy logra identificar con precisión las ecuaciones dinámicas del sistema cuando al utilizar una única trayectoria en el proceso de reconstrucción (el estudio se puede extender usando múltiples trayectorias en el algoritmo), pero no logra reproducir la dinámica completa de órbitas caóticas.

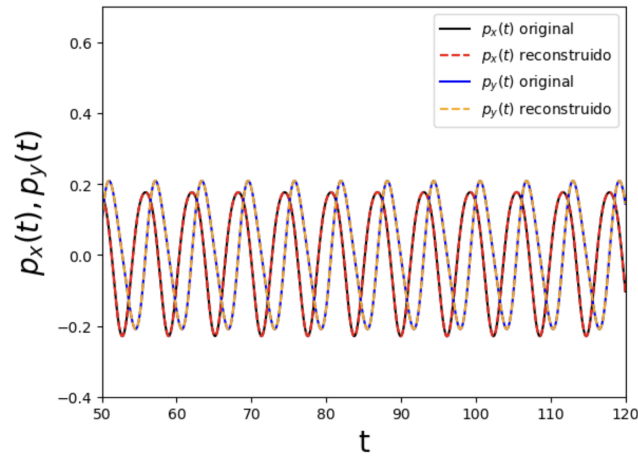
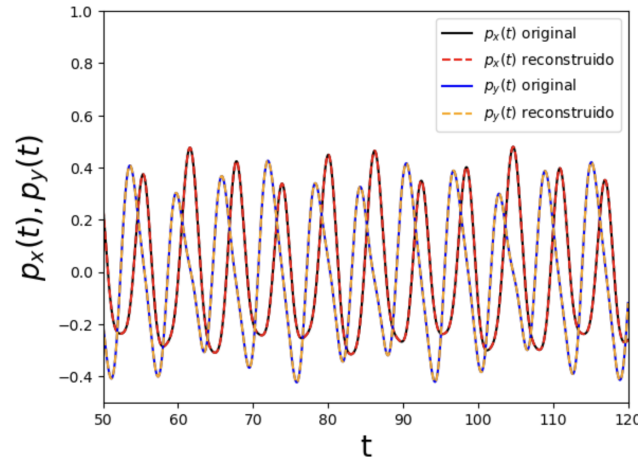
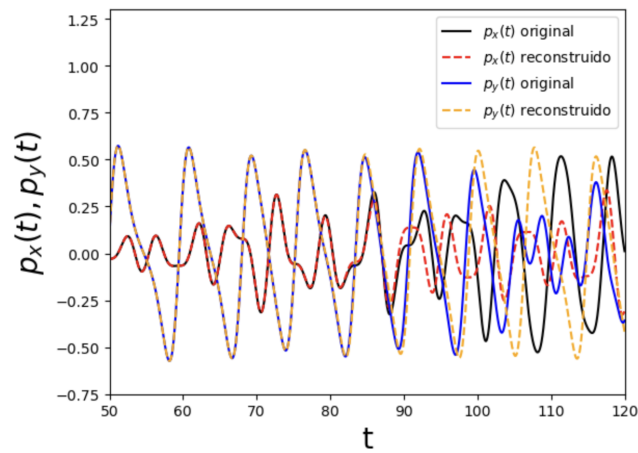
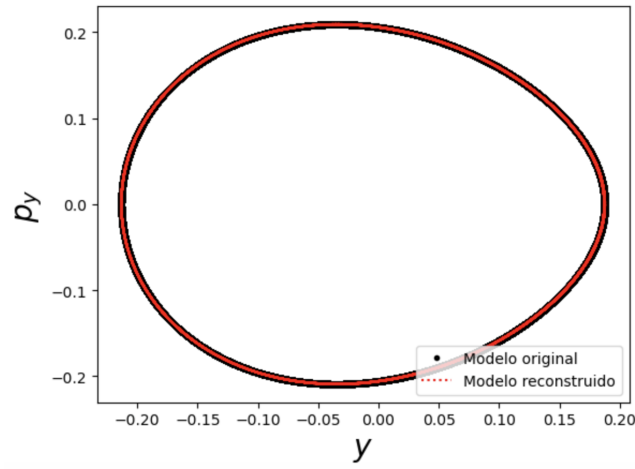
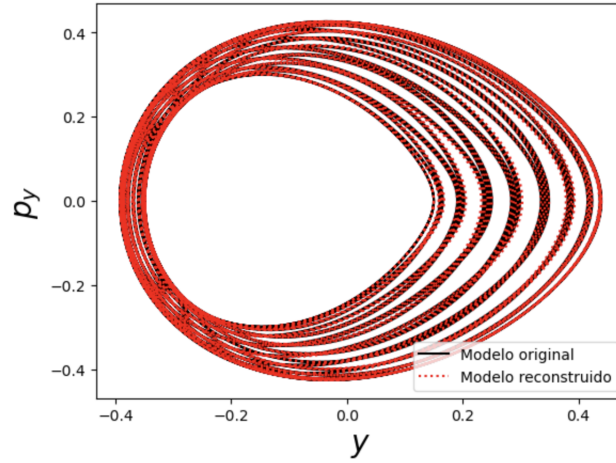
(a) Órbita periódica, $E = \frac{1}{4}E_c$ (b) Órbita cuasiperiódica, $E = \frac{3}{4}E_c$ (c) Órbita caótica, $E = \frac{99}{100}E_c$

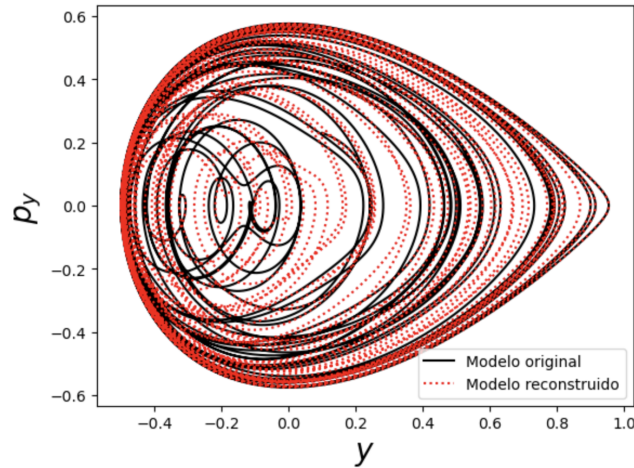
Figura 7.10: Para valores $dt = 1 \times 10^{-4}$ se muestra una comparación entre la serie temporal original de las variables $p_x(t)$, $p_y(t)$ y las que se obtienen simulando las ecuaciones reconstruidas con el algoritmo SINDy. Para las órbitas periódicas y cuasiperiódicas, en este caso particular del valor en Δt , el modelo reconstruido reproduce el comportamiento durante todo el intervalo de tiempo; mientras que en el caso de la órbita caótica (en la región cercana de energía crítica E_c), a pesar de reconstruir las ecuaciones de Hamilton correctamente, la serie temporal diverge de la original durante su evolución.



(a) Órbita periódica, $E = \frac{1}{4}E_c$



(b) Órbita cuasiperiódica, $E = \frac{3}{4}E_c$



(c) Órbita caótica, $E = \frac{99}{100}E_c$

Figura 7.11: Para las condiciones iniciales de las series temporales mostradas en la Figura 7.10 se muestra la comparación en el plano y vs p_y de las órbitas del modelo original y las que se obtienen con el modelo reconstruido por SINDy. En el caso periódico y cuasiperiódico las órbitas coinciden, mientras que en el caso caótico como lo muestra la serie temporal, existe una perdida de precisión a tiempos largos.

Soluciones periódicas aproximadas de forma analítica

Anteriormente se mostró que el algoritmo de aprendizaje automatizado conduce a expresiones analíticas aproximadas para los tres tipos de trayectorias e incluso en el caso en que se conozca el modelo exacto y, por lo tanto, no sea realmente necesario reconstruir la ecuación dinámica, aún se puede aprovechar SINDy para derivar soluciones aproximadas de manera explícita.

Sea entonces el caso $N = 4$ con energía $\frac{1}{4}E_c$, donde, para un valor $dt = 2 \times 10^{-5}$ aparecen términos extra en las ecuaciones reconstruidas de la condición inicial periódica pero las herramientas de medición del modelo ϵ_{MSE} y R^2 indican que el modelo es adecuado para capturar la dinámica; lo cual no ocurre para los otros tipos de órbitas. Las ecuaciones obtenidas con SINDy, con precisión de 5 cifras significativas, son:

$$\begin{aligned}
 \dot{x}_{\text{SINDy}} &= p_x , \\
 \dot{y}_{\text{SINDy}} &= p_y , \\
 \dot{p}_{x\text{SINDy}} &= 0,00322 y - 1,00132 x - 0,75788 x y^2 \\
 &\quad + 2,81860 y p_x p_y - 1,88935 y x^2 - 1,19774 y p_x^2 \\
 &\quad - 1,16865 x p_x p_y - 0,25909 x^3 + 0,49646 x p_x^2 , \\
 \dot{p}_{y\text{SINDy}} &= -y + 2,48528 y x^2 - 0,00042 x p_x p_y \\
 &\quad - 1,51478 x^3 + 0,00012 x p_x^2 .
 \end{aligned} \tag{7.33}$$

Conociendo el modelo exacto, el lado izquierdo de las ecuaciones (7.33) se pueden separar en la suma de términos extra y exactos de la forma,

$$S_{\tau, \text{SINDy}} = S_{\tau, \text{exact}} + S_{\tau, \text{extra}} . \tag{7.34}$$

Claramente no hay términos extra para las ecuaciones $\dot{x}_{\text{SINDy}}, \dot{y}_{\text{SINDy}}$; mientras que para las ecuaciones de los momentos, realizando la separación de términos, se tiene

$$\begin{aligned}
 S_{\dot{p}_x, \text{extra}} &= 0,00322 y - 0,00132 x - 0,75788 x y^2 - y^3 \\
 &\quad + 2,81860 y p_x p_y + 1,11065 y x^2 - 1,19774 y p_x^2 \\
 &\quad - 1,16865 x p_x p_y - 0,25909 x^3 + 0,49646 x p_x^2 , \\
 S_{\dot{p}_y, \text{extra}} &= 2,48528 y x^2 - 3 x y^2 - 0,00042 x p_x p_y \\
 &\quad - 0,51478 x^3 + 0,00012 x p_x^2 .
 \end{aligned} \tag{7.35}$$

Ahora, evaluando (7.35) sobre la trayectoria numérica original de la condición inicial periódica en el Cuadro 7.6, la contribución de $S_{\dot{p}_x, \text{extra}}$, $S_{\dot{p}_y, \text{extra}}$ es prácticamente cero, Fig. 7.12.

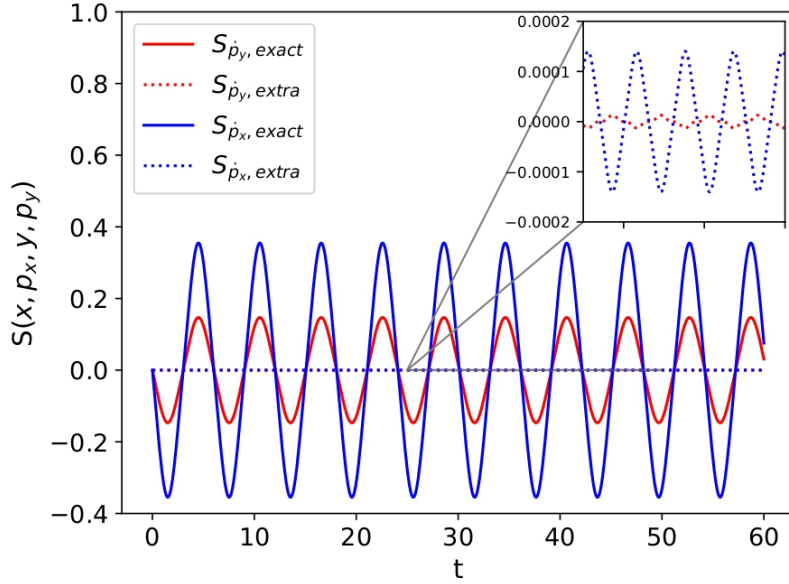


Figura 7.12: Caso $N = 4$: Series de tiempo para los términos en (7.34), de la parte exacta y extra de las ecuaciones obtenidas con SINDy. Estos términos son evaluados sobre la trayectoria periódica en Fig. 7.7 (d). La variación de la parte extra es del orden $\leq 10^{-4}$.

En particular, de la segunda ecuación en (7.35), despreciando los términos con coeficientes de orden $\approx 10^{-4}$, se obtiene la relación analítica de forma aproximada:

$$y = 0,41x, \quad (7.36)$$

que corresponde a la forma de la trayectoria periódica, mostrada en la Fig. 7.7 (d). Por consiguiente, para valores Δt extremadamente reducidos, SINDy puede producir términos no representativos durante la reconstrucción de las ecuaciones fundamentales de movimiento.

Modelo de ECG como sistema dinámico

En esta última parte del trabajo se explora el uso de SINDy para reconstruir un modelo que describe la generación de señales de electrocardiogramas, las cuales transicionan a órbitas caóticas al realizar la variación de cierto parámetro en las ecuaciones dinámicas, este caos está relacionado con algunos tipos de arritmias las cuales pueden ser mortales sin una intervención médica. Así mismo, se plantea el estudio de la habilidad de SINDy para recuperar las ecuaciones y la dinámica a partir de una sola variable de estado medida.

8.1. Generación de un electrocardiograma (ECG)

El bombeo de sangre en el corazón inicia mediante la actividad del nodo sinuauricular (SA), este impulso eléctrico viaja a través de los músculos atriales hasta el nodo atrioventricular (AV) hasta que finalmente los impulsos se propagan a lo largo del complejo de His-Purkinje, causando la contracción de los músculos ventriculares; este proceso conduce al suministro de sangre a todo el cuerpo [30]. Los eventos que ocurren cada ciclo cardíaco reflejan el flujo de corriente iónica, lo que genera un potencial eléctrico variable en el tiempo en la superficie de la piel, el cual es registrado en un electrocardiograma (ECG) y ha sido utilizado para detectar anomalías del corazón, denominadas como arritmias. Una de estas anomalías es la fibrilación ventricular (VF), la cual es maligna sin intervención médica y puede ocasionar la muerte en un instante. Dado que la VF se distingue por un estado de contracción arrítmica irregular de los músculos ventriculares, se ha señalado como un proceso estocástico y errático; estu-

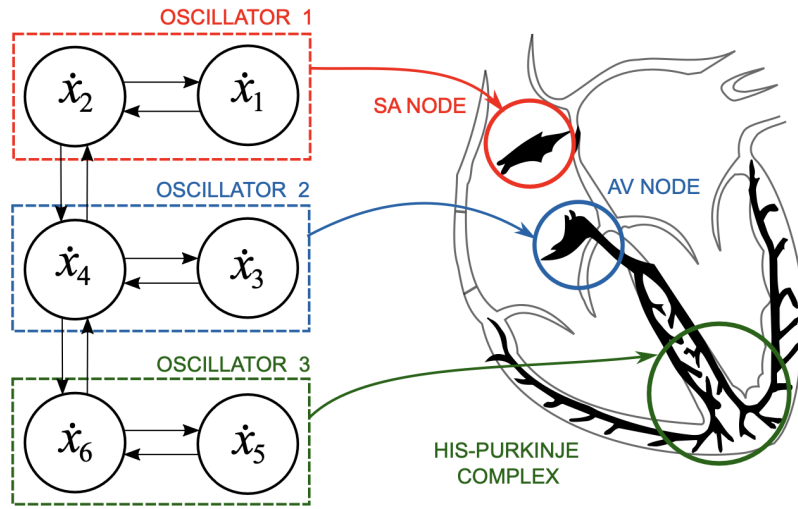


Figura 8.1: Diagrama de los marcapasos cardíacos naturales relacionados a las variables $x_1 - x_6$ del modelo discretizado en [32]. La acción de los nodos SA, AV y el complejo His-Purkinje son representados por tres osciladores.

dios experimentales sugieren la presencia de caos, resultado de dinámicas deterministas no lineales [31]. Sin embargo, la teoría de sistemas dinámicos no lineales se reconoce como una herramienta útil que puede contribuir a la comprensión de los mecanismos subyacentes de arritmias letales y enfermedades cardíacas.

La dinámica cardíaca ha sido objeto de extensa investigación, lo que ha resultado en el desarrollo de modelos matemáticos con diversos grados de complejidad. El que se utiliza para este trabajo se encuentra descrito en el estudio realizado por Quiroz et al.(2019)[32] donde se presenta un modelo fundamentado en el mecanismo de reacción-difusión [33],[34] con el propósito de simular la generación de señales ECG. Dicho modelo se discretiza de manera que se genera un conjunto de tres osciladores no lineales acoplados, formados por las variables x_i , $i = 1, 2, 3, 4, 5, 6$, los cuales representan la actividad eléctrica del corazón. Estos tres osciladores pueden asociarse a los marcapasos naturales presentes en el corazón como el nodo SA, el nodo AV y el complejo de His-Purkinje, Fig. 8.1, por lo tanto las señales del electrocardiograma pueden ser simuladas como una señal combinada de las variables de estos osciladores. Tras consideraciones específicas, las variables de los tres osciladores mostrados en la Fig. 8.1 se reducen, dejando la dinámica como función de las variables x_1, x_2, x_3, x_4 , por lo tanto el modelo

se compone de las siguientes ecuaciones diferenciales no lineales acopladas,

$$\dot{x}_1 = x_1 - x_2 - Cx_1x_2 - x_1x_2^2, \quad (8.1)$$

$$\dot{x}_2 = Hx_1 - 3x_2 + Cx_1x_2 + x_1x_2^2 + \beta(x_4 - x_2), \quad (8.2)$$

$$\dot{x}_3 = x_3 - x_4 - Cx_3x_4 - x_3x_4^2, \quad (8.3)$$

$$\dot{x}_4 = Hx_3 - 3x_4 + Cx_3x_4 + x_3x_4^2 + 2\beta(x_2 - x_4). \quad (8.4)$$

Donde H representa un parámetro de control, β la constante de acoplamiento del Laplaciano discreto y C la intensidad de los términos cúbicos y cuadráticos. Resultados experimentales [35] sugieren que la fibrilación ventricular es una forma de caos espacio temporal, en [32] se muestra que el sistema formado por las ecuaciones (8.1)-(8.4) puede exhibir este comportamiento con la variación del parámetro H ; así mismo se presenta una discusión sobre la riqueza en la variedad de dinámicas del sistema. Una familia de estados estacionarios se forma para valores de $H > 9$, mientras que al disminuir el valor de H se origina una bifurcación de Hopf (HB) en $H = 8,779267$, dando lugar a una rama caracterizada por ciclos límite estables. Esta rama experimenta una pérdida de estabilidad al alcanzar un punto de duplicación de Periodo (PD) en $H = 2,742524$, formando órbitas cuasiperiódicas y finalmente, atraviesa una transición a toroide (TR) en $H = 2,72972$ correspondientes a presencia de caos. Esta descripción puede observarse en la Fig. 8.2, donde el ciclo límite inicial se va deformando simultáneamente con la variación de H hasta obtener la cuasiperiodicidad y pequeñas perturbaciones transforman la órbita a un movimiento caótico o un atractor extraño.

Finalmente, se considera, como primera aproximación, que la señal ECG es generada por la combinación lineal entre las variables de estado x_i de forma,

$$ECG(t) = \alpha_1x_1 + \alpha_2x_2 + \alpha_3x_3 + \alpha_4x_4. \quad (8.5)$$

A continuación, se realiza la reconstrucción del modelo descrito por las ecuaciones (8.1)-(8.4) usando la órbita del ciclo límite, el estado con doble periodo y el estado caótico, con los mismos parámetros establecidos en [32], $C = 1,35$ y $\beta = 4$.

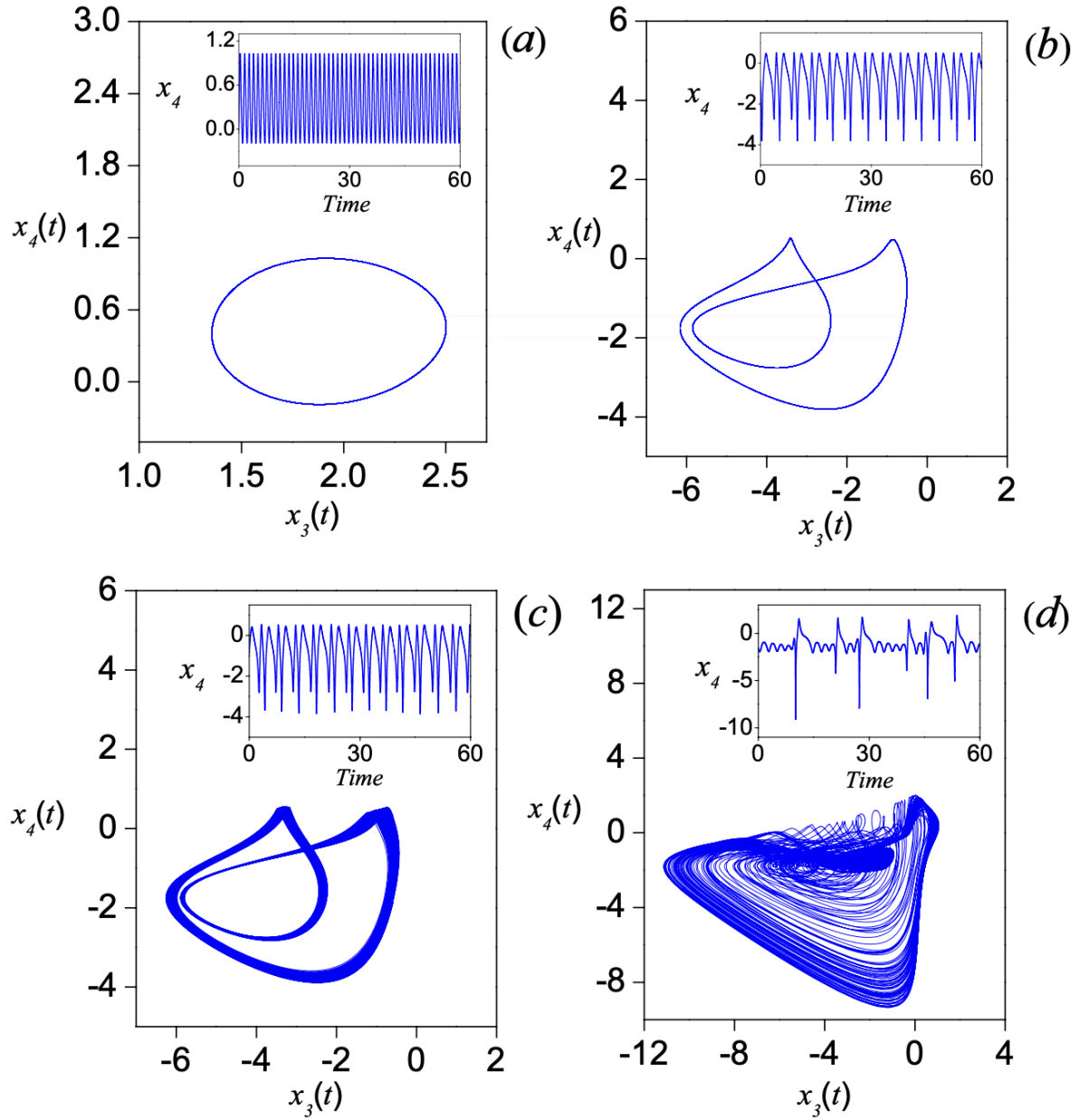


Figura 8.2: Espacio fase x_3 vs x_4 y series temporales para los siguientes valores del parámetro de control H después de $t = 500$ segundos: (a) $H = 7$ estado estacionario, (b) $H = 2,74$ bifurcación de Hopf, (c) $H = 2,72972$ estado con doble periodo, (d) $H = 2,7126$ estado caótico. En todos los casos se toman los valores $C = 1,35$ y $\beta = 4$. [32].

8.2. Reconstrucción de ecuaciones dinámicas con SINDy

Se utiliza el sistema dinámico (no-Hamiltoniano) descrito por las ecuaciones (8.1)-(8.4), que conforman un modelo para la generación de una señal de electrocardiograma (ECG), para probar la habilidad del algoritmo SINDy en presencia de caos generado al realizar variaciones del parámetro H el cual ocasiona transiciones en la naturaleza de las órbitas. Para las órbitas de la Fig. 8.2(a)-(d), se reproduce de forma sistemática la reconstrucción de las ecuaciones dinámicas usando SINDy con los valores del parámetro de control $H = 7$, $H = 2,74$, $H = 2,72972$, $H = 2,7126$, para los cuales se obtuvo las series temporales y los retratos en el espacio fase, Fig. 8.3, usando las siguientes condiciones iniciales:

H	$(x_{1_0}, x_{2_0}, x_{3_0}, x_{4_0})$
7	$(0, 0, 0, 1, 0)$
2,74	$(0, 0, 0, 1, 0)$
2,72972	$(3,75127454, 0,42153287, -2,67657483, -0,72261479)$
2,7126	$(3,75127454, 0,42153287, -2,67657483, -0,72261479)$

Las series temporales se generan mediante integración numérica (RK-Fehlberg [12]) de las ecuaciones (8.1)-(8.4) para diferentes valores del paso de integración dt , con las cuales se alimenta el algoritmo de SINDy, junto con el uso del optimizador STLQ y el respectivo hiperparámetro aproximado con frente de Pareto (Fig. 7.8, 7.9). Los resultados de las ecuaciones obtenidas con SINDy se muestran en el Cuadro 8.1-8.3.

Usando el parámetro de error relativo $\Delta C_\tau(\sigma)$, presentado en la Sección 7.3, se obtienen funciones gradualmente crecientes que muestran la pérdida de precisión entre el modelo exacto y el reconstruido con SINDy a medida que incrementa el parámetro dt ; a manera de ilustración se muestran las curvas correspondientes para los términos $\sigma = x_1, x_2, x_1x_2$, $\sigma = x_2, x_4, x_3x_4, x_3x_4^2$ y $\sigma = x_3$ de las ecuaciones $\dot{x}_1, \dot{x}_4, \dot{x}_3$, respectivamente, en la Fig. 8.4 (a)-(d). En general, el algoritmo de aprendizaje automático reconstruye las ecuaciones dinámicas del modelo para valores infinitesimales en dt , como sucedía en el caso del sistema de Hénon-Heiles.

En el caso de la órbita periódica con el parámetro $H = 7$, se nota que aún para valores cercanos en dt del mismo orden de magnitud (10^{-4}), se producen términos extras en las ecuaciones dinámicas como se indica en el Cuadro 8.1. Lo cual indica que

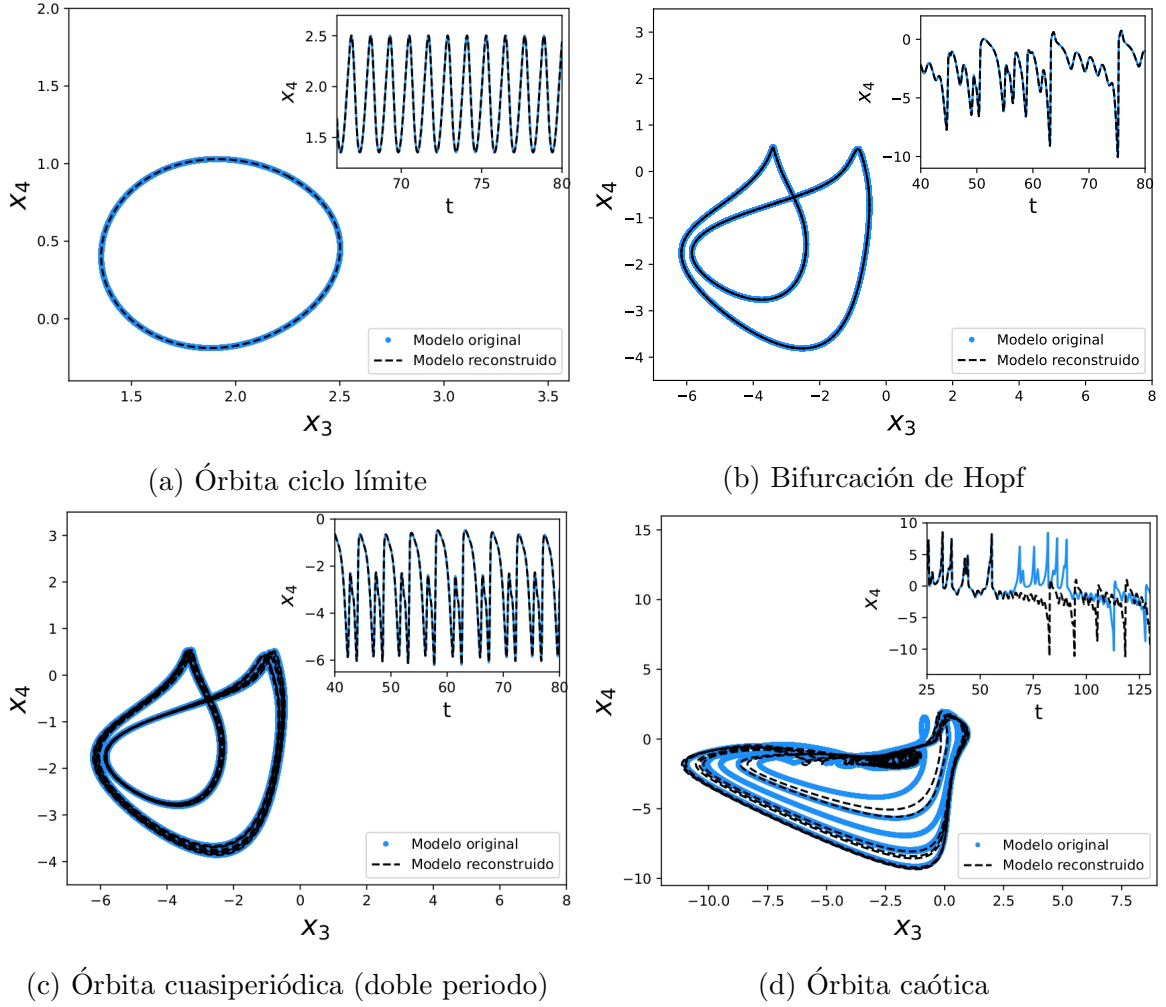


Figura 8.3: Comparación entre la reconstrucción obtenida con SINDy para las órbitas presentadas en la Fig. 8.2. Para el caso caótico en (d) se pierde la precisión de la reconstrucción a cierto tiempo.

el límite dt_c no es necesariamente un valor grande y es independiente para cada tipo de dinámica. Al conocer el modelo exacto, se realiza la separación de las ecuaciones con términos extra, en la forma

$$S_{\tau, \text{SINDy}} = S_{\tau, \text{exact}} + S_{\tau, \text{extra}} ; \quad (8.6)$$

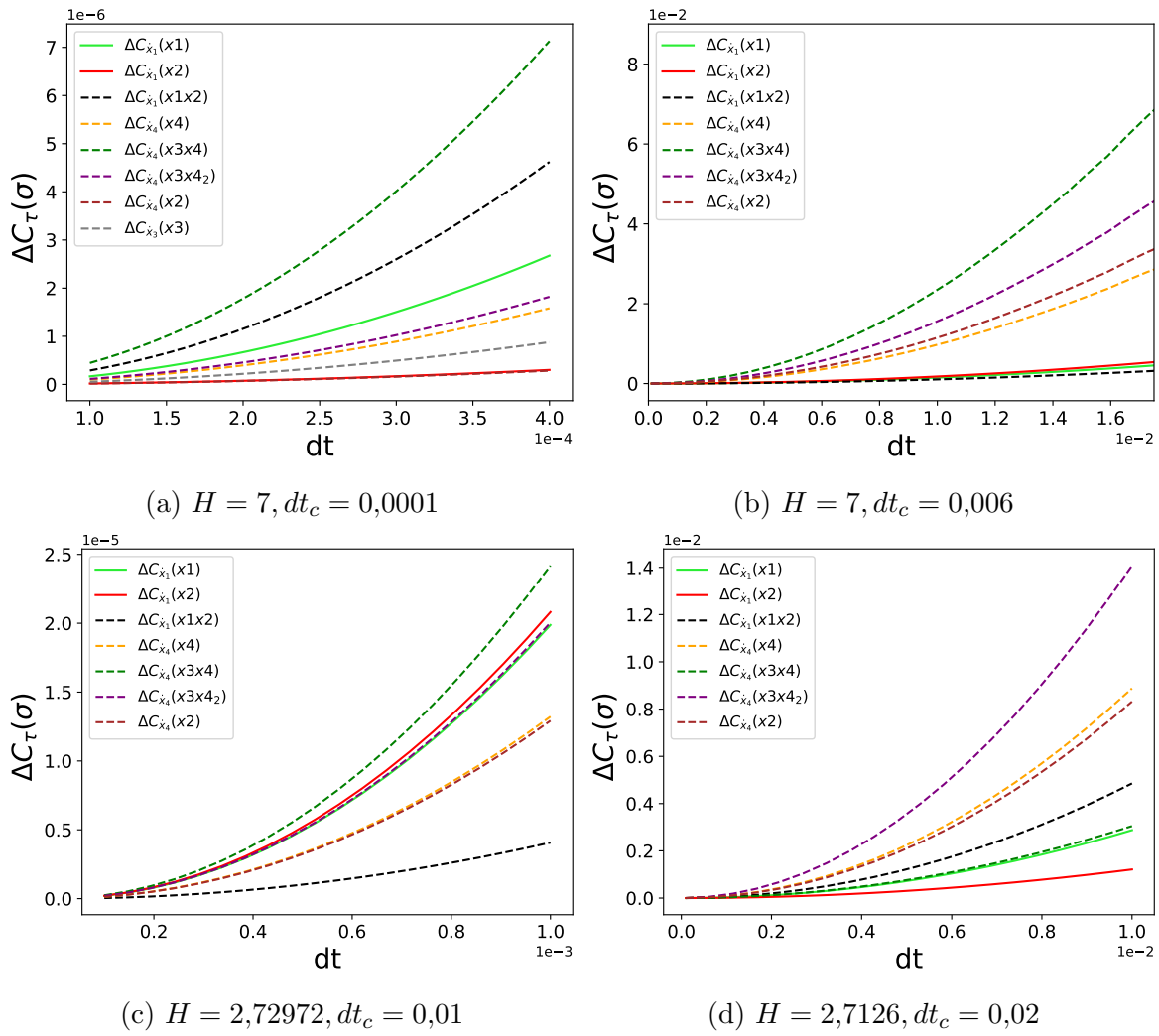


Figura 8.4: Error relativo de los coeficientes $C_\tau(\sigma)$ en función del incremento del paso de integración dt . Debido al incremento del parámetro H en las ecuaciones del modelo, el límite dt_c emerge gradualmente a medida que el sistema exhibe comportamiento caótico.

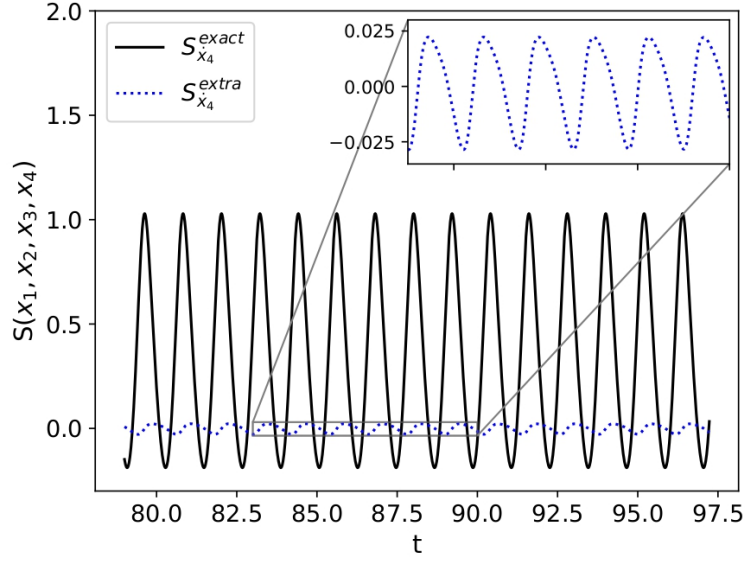


Figura 8.5: Evaluación de los términos $S_{x_4,extra}$ sobre la trayectoria numérica original, apreciando una contribución del orden 10^{-2} a la trayectoria original, lo cual ocasiona una pérdida de precisión en el modelo obtenido con SINDy.

tomando el caso $dt = 0,0006$ del Cuadro 8.1, se obtiene:

$$\begin{aligned}
 S_{x_4,extra} = & -1,281x_3 + 3,43x_4 - 3,221x_2 - 0,005x_3x_4^2 - 1,35x_3x_4 \\
 & + 1,945x_1 - 0,425x_1x_3 - 0,153x_2^2 - 2,144x_2x_4 - 0,587x_1^2x_4 \\
 & - 0,34x_1x_2x_3 - 0,497x_1x_3x_4 + 0,542x_1x_4^2 - 0,693x_2^2x_4 .
 \end{aligned} \tag{8.7}$$

Se evalúa (8.7) sobre la trayectoria numérica original, de donde se obtiene que la contribución es del orden $\approx 10^{-2}$, Fig. 8.5, lo cual incrementa el error cuadrático medio y decrece la precisión del modelo como lo indica R^2 en el Cuadro 8.3. Reafirmando así que para el límite dt_c los modelos que se obtienen con el algoritmo SINDy comienzan a perder precisión por la presencia de términos extra en las ecuaciones dinámicas.

8.3. Construcción de modelos a partir de una variable de estado

En las secciones anteriores, usando un sistema Hamiltoniano y dinámico se mostró la habilidad de SINDy en la reconstrucción de las ecuaciones dinámicas ya conocidas,

a partir de series temporales para valores infinitesimales del paso de integración dt , independientemente de la naturaleza de la dinámica. Ahora, surge una interrogante crucial: ¿Se pueden construir ecuaciones de movimiento y replicar la dinámica de un sistema si solo se dispone de mediciones para una sola variable de estado?; para abordar este cuestionamiento de forma introductoria, se realiza una reconstrucción simple con SINDy y el teorema de Takens (Sec. 8.3.1), del atractor de Lorenz, usando la serie temporal de la variable x . Esta idea puede ser fundamental para encontrar ecuaciones de sistemas en los que se desconoce la estructura dinámica completa. En relación al modelo propuesto para la descripción de señales ECG, sus implicaciones exceden el alcance de esta investigación, por lo que se propone como una vía de estudio para trabajos futuros.

8.3.1. Teorema de Takens

Supongamos que se desea estudiar un sistema en el que cada estado esta determinado por las variables $x_1(t), x_2(t), \dots, x_n(t)$, de las cuales solo se tienen las mediciones en una serie de tiempo para una sola variable, por ejemplo x_1 . La estructura geométrica del espacio fase puede recuperarse usando el método de coordenadas retardadas o mejor conocido como el teorema de Takens [36], el cual consiste en graficar en un espacio $\mathcal{R}^{2d+1}$, con d la dimensión del espacio de retardo, las variables de estado s_n construidas con x_1 de la siguiente forma,

$$s_1 = x_1(t) \quad , \quad s_2 = x_1(t + \tau) \quad , \quad s_3 = x_1(t + 2\tau), \quad \dots \quad , s_n = x_1(t + (d - 1)\tau) \quad , \quad (8.8)$$

donde τ es el parámetro del desplazamiento temporal, el cual puede establecerse de forma aleatoria mientras se reproduzca la misma dinámica que en el espacio fase original o utilizando el método del promedio de información mutua (AMI) [37] para calcular el valor más adecuado; también se puede encontrar la dimensión del nuevo espacio fase utilizando el método de falsos vecinos más cercanos (FNN) [38]. La dinámica que se obtiene con los estados s_n utilizando este teorema se expresa como un homeomorfismo entre espacios fase, lo cual significa que uno se obtiene a través de una deformación del otro y se destaca que, independientemente de sus formas individuales, comparten las mismas características o propiedades.

8.3.2. Atractor de Lorenz mediante el método de retardo temporal

A manera de ilustración del uso del Teorema de Takens en conjunto con SINDy, se considera el atractor de Lorenz, sistema que presenta caos, descrito brevemente en la Sección(6.4), se realiza la integración numérica (con el método RK-Fehlberg [12]) de las ecuaciones (6.21)-(6.23) para obtener las series temporales de las variables x , y , z , el plano y vs z se muestra en la Fig. 8.6(a). Al considerar únicamente los datos en la variable x se reconstruye la dinámica en el espacio fase usando el teorema de Takens calculando las series de tiempo de las variables s_1 , s_2 , s_3 , con el parámetro de retardo $\tau = 0,1$, Fig.8.6(b), seleccionado por ser el valor en el cual se obtiene una estructura fase similar a la original .

Utilizando las series temporales generadas para las variables s_n , con un paso de integración pequeño, $dt = 0,0001$, se busca obtener las ecuaciones que describan la dinámica en este nuevo espacio mediante el uso de SINDy, y se comparan con las ecuaciones conocidas. Durante el proceso de construcción del modelo con el algoritmo de aprendizaje automático, se enfrentan ciertas complicaciones para obtener un resultado parsimonioso, en particular en el ajuste del hiperparámetro λ para valores $\lambda > 0,01$. Después de un análisis detallado, se determina que se obtiene un modelo satisfactorio con $\lambda = 0,0128$, lo que sugiere que el valor del hiperparámetro puede ser lo suficientemente pequeño para garantizar una aproximación adecuada del modelo. Las ecuaciones del modelo parsimonioso que se obtiene con SINDy son de la forma,

$$\left\{ \begin{array}{l} \dot{s}_{1SINDy} = -0,0391 - 18,308s_1 + 23,686s_2 - 5,058s_3 + 0,038s_1^3 - 0,061s_1^2s_2 \\ \quad \quad \quad + 0,033s_1^2s_3 + 0,067s_1s_2^2 - 0,046s_1s_2s_3 - 0,035s_2^3 \\ \dot{s}_{2SINDy} = -3,941s_1 - 1,880s_2 + 5,230s_3 - 0,037s_1s_2^2 + 0,026s_1s_2s_3 + 0,021s_2^3 \\ \dot{s}_{3SINDy} = 4,360s_1 - 20,826s_2 + 16,968s_3 + 0,029s_1s_2^2 - 0,032s_2s_3^2 \end{array} \right. \quad (8.9)$$

Al evaluar el modelo obtenido con respecto a la órbita en las variables s_n , se observa que el error cuadrático medio alcanza un valor de $\epsilon_{MSE} = 134,6812$ y coeficiente de correlación $R^2 = -1,18$. Aunque el valor de ϵ_{MSE} no es el más bajo entre los modelos

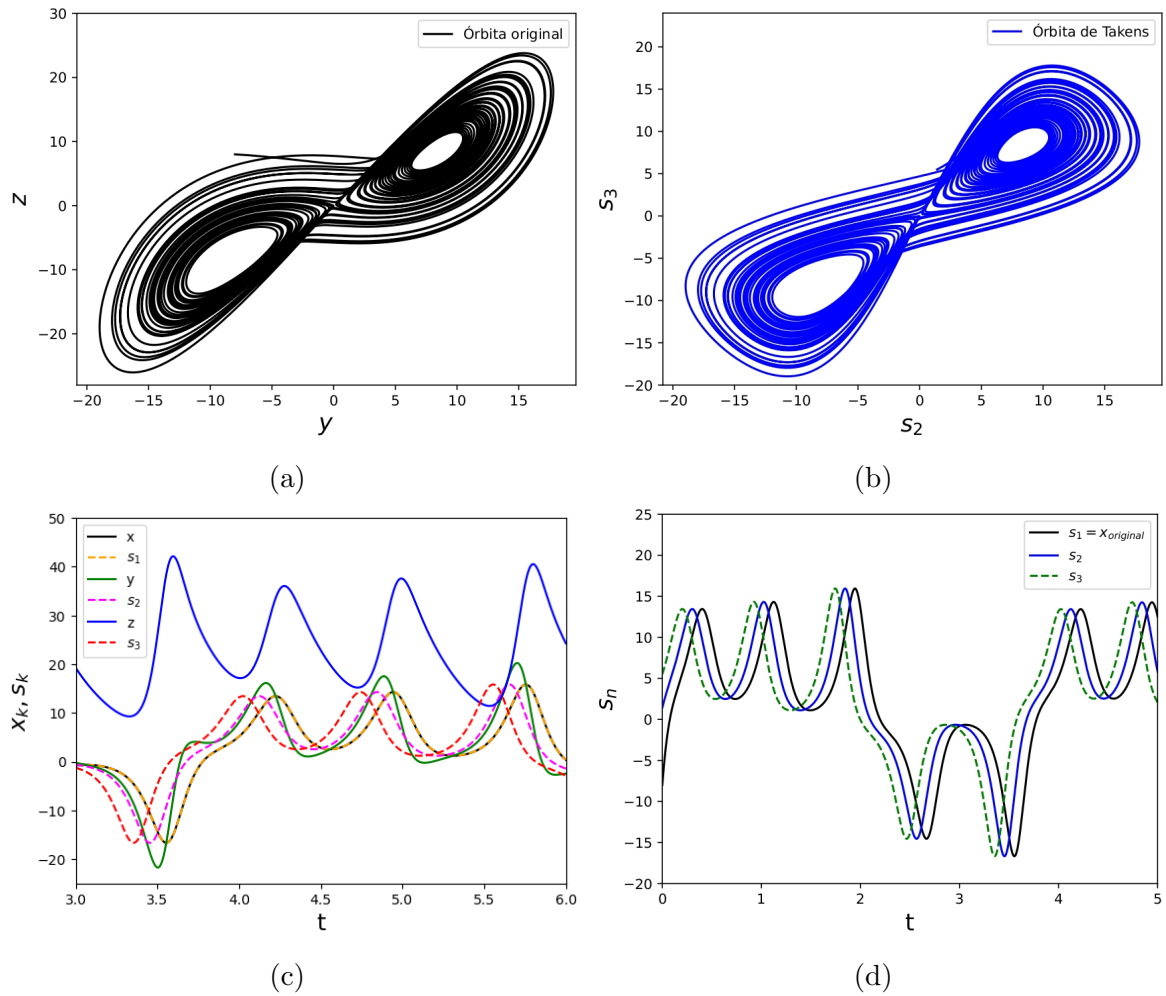


Figura 8.6: Usando el teorema de Takens con el valor del parámetro de retardo $\tau = 0,1$ para la serie temporal de la variable x en el sistema del atractor de Lorenz, se construyen las series temporales s_2 , s_3 ; el plano correspondiente a estas variables en (b) muestra un difeomorfismo del plano original y vs z en (a). Una sección de las series temporales de ambas representaciones se muestran en (c) y los retardos de la variable x en (d).

calculados, se destaca por su capacidad para aproximar de manera adecuada la dinámica en el espacio de fase y por generar ecuaciones con menos términos en comparación con los demás.

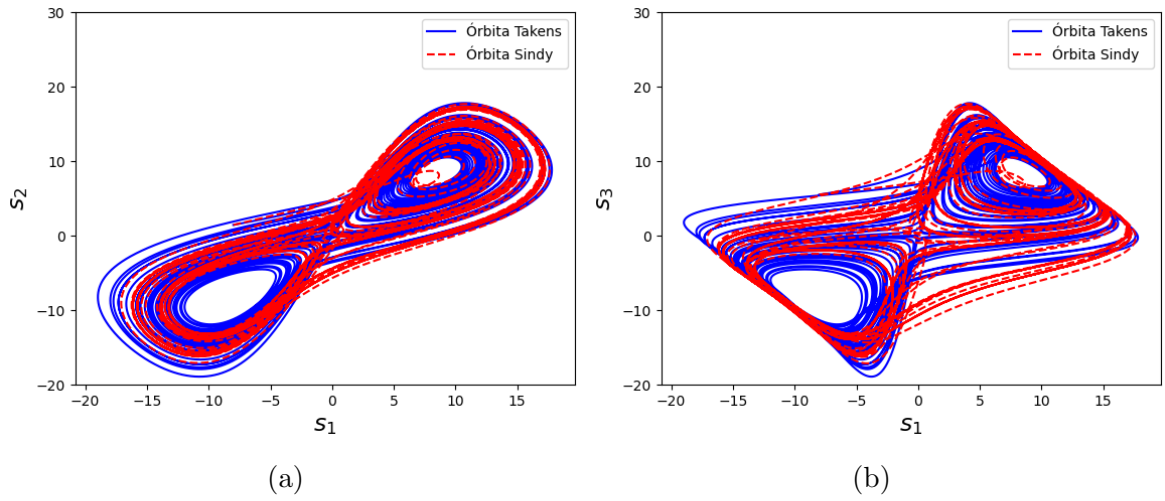


Figura 8.7: Se realiza una comparación entre una órbita del atractor de Lorenz en los planos s_1 vs s_2 y s_1 vs s_3 de las variables s_i , obtenidas utilizando el Teorema de Takens (Órbita Takens), y la órbita generada a partir de las ecuaciones (8.9) mediante el empleo del algoritmo SINDy (Órbita Sindy).

dt	Modelo SINDy	ϵ_{MSE}	R^2
0.0001	$\dot{x} = 1,00000019x_1 - 0,9999x_2$ $\dot{x}_2 = 7,00000x_1 + -7,00000x_2$ $+4,00000x_4 + 1,35000x_1x_2 + 1,00000x_1x_2^2$ $+1,00000x_1x_2^2$ $\dot{x}_3 = 0,9999x_3 - 1,00x_4$ $-1,34999x_3x_4 - 0,9999x_3x_4^2$ $-1,34999957x_1x_2 - 0,99999981x_1x_2^2$	$1,56417 \times 10^{-10}$	0.9999999990744667
0.0006 dt_c	$\dot{x}_1 = 1,0001x_1 - 0,99999925x_2$ $-1,34998x_1x_2 - 0,99999x_1x_2^2$ $\dot{x}_2 = 6,99998635x_1 - 6,99998666x_2$ $+3,99999443x_4 + 1,34999433x_1x_2$ $0,99999636x_1x_2^2$ $\dot{x}_3 = 0,99999788x_3 - 1,00001233x_4$ $-1,34999826x_3x_4 - 0,99998768x_3x_4^2$ $\dot{x}_4 = 1,94496033x_1 + 4,7788637x_2$ $+5,71888x_3 - 7,56963x_4$ $+0,40504x_1x_2 + -0,42508x_1x_3$ $-0,15337x_2^2 - 2,14370x_2x_4$ $-0,58724x_1^2x_4 - 0,34006x_1x_2x_3$ $-0,49664x_1x_3x_4 + 0,54219x_1x_4^2$ $-0,69335x_2^2x_4 + 0,99455x_3x_4^2$	0.00019819	0.9988271039323753
0.001	$\dot{x}_1 = 1,00002x_1 + -1,00000x_2$ $-1,34996x_1x_2 - 0,99998x_1x_2^2$ $\dot{x}_2 = 4,30688x_1 - 2,51359x_2$ $+1,42321x_3 - 0,37870x_1x_2$ $+1,14701x_2^2 - 1,14637x_2x_4$ $+1,08863x_4^2 - 0,24363x_1^2x_2$ $+0,82534x_1x_2^2$ $\dot{x}_3 = 0,99999x_3 - 1,00003x_4$ $-1,35000x_3x_4 - 0,99997x_3x_4^2$ $\dot{x}_4 = 0,93922x_1 + 5,82217x_2$ $+6,01028x_3 - 8,77573x_4$ $-0,40465x_1x_3 + 0,77441x_1x_4$ $-0,12801x_2^2 + 0,74999x_2x_3$ $-1,34281x_2x_4 + 0,37296x_3^2$ $+0,29562x_4^2 - 0,01631x_1x_2x_3$ $-0,26611x_1x_3x_4 + 0,29371x_1x_4^2$ $-0,40786x_2^2x_4 + 1,11495x_3x_4^2$	0.04091876	0.75774399

Cuadro 8.1: Reconstrucción de las ecuaciones dinámicas del caso ciclo límite ($H = 7$) para formar la señal ECG usando diferentes pasos de integración dt .

dt	Modelo SINDy	ϵ_{MSE}	R^2
0.0001	$\dot{x}_1 = 0,9999998x_1 - 1,00000021x_2$ $-1,34999995x_1x_2 - 0,9999997x_1x_2^2$ $\dot{x}_2 = 2,72971989x_1 - 6,99999921x_2$ $+3,99999967x_4 + 1,34999977x_1x_2$ $0,99999983x_1x_2^2$ $\dot{x}_3 = 1,00000005x_3 - 1,00000037x_4$ $-1,34999963x_3x_4 - 0,99999979x_3x_4^2$ $\dot{x}_4 = 7,99999897x_2 + 2,72971965x_3$ $-10,99999855x_4 + 1,34999967x_3x_4$ $+0,9999998x_3x_4^2$	$5,982923 \times 10^{-6}$	0.9999954695794566
0.01 dt_c	$\dot{x}_1 = 0,998x_1 - 1,002x_2$ $-1,349x_1x_2 + -0,997x_1x_2^2$ $\dot{x}_2 = 2,729x_1 - 6,992x_2$ $+3,997x_4 + 1,348x_1x_2$ $+0,998x_1x_2^2$ $\dot{x}_3 = 1,000x_3 - 1,004x_4$ $-1,346x_3x_4 - 0,998x_3x_4^2$ $\dot{x}_4 = 0,2441 + 1,077x_1$ $+1,447x_3 - 1,921x_4$ $+0,780x_1x_2 - 0,626x_1x_4$ $-0,507x_2x_3 + 0,611x_2x_4$ $+1,218x_3x_4 + 0,638x_1x_2x_4$ $-0,408x_2x_3x_4 - 0,372x_2x_4^2$ $+0,892x_3x_4^2$	245,9437	-254.5574

Cuadro 8.2: Reconstrucción de las ecuaciones dinámicas de una señal cuasiperiódica ($H = 2,72972$) para formar la señal ECG usando diferentes pasos de integración dt .

dt	Modelo SINDy	ϵ_{MSE}	R^2
0.0001	$\dot{x}_1 = 0,99999981x_1 - 1,00000015x_2$ $-1,34999934x_1x_2 - 0,99999972x_1x_2^2$ $\dot{x}_2 = 2,71259944x_1 - 6,9999989x_2$ $+3,999x_4 + 1,349999x_1x_2$ $+0,9999x_1x_2^2$ $\dot{x}_3 = 0,999998x_3 - 1,000002x_4$ $-1,35000032x_3x_4 - 0,9999986x_3x_4^2$ $\dot{x}_4 = 7,9999932x_2 + 2,712599x_3$ $-10,9999x_4 + 1,3499995x_3x_4$ $+0,9999985x_3x_4^2$	0.89134811	0.51115
0.02 dt_c	$\dot{x}_1 = 0,99240x_1 - 1,00590x_2$ $-1,34552x_1x_2 - 0,98863x_1x_2^2$ $\dot{x}_2 = 2,669x_1 - 6,912x_2$ $+3,96378x_4 + 1,34032x_1x_2$ $+0,98637x_1x_2^2$ $\dot{x}_3 = -0,54250x_2 + 1,06814x_3$ $-0,47692x_2^2 - 0,23978x_2x_3$ $+0,83499x_2x_4 - 0,96971x_3x_4$ $-0,91093x_3x_4^2$ $\dot{x}_4 = 8,25893x_2 + 2,45526x_3$ $-11,28912x_4 - 0,36870x_2^2$ $+0,66267x_2x_4 + 0,90956x_3x_4$ $-0,30903x_4^2 + 0,84232x_3x_4^2$	9,646461	-3,18907

Cuadro 8.3: Reconstrucción de las ecuaciones dinámicas de una señal caótica ($H = 2,2,7126$) para formar la señal ECG usando diferentes pasos de integración dt .

Conclusiones

En este trabajo se analizó, usando el sistema (caótico) generalizado de Hénon-Heiles como sistema de prueba, la aplicación del algoritmo de aprendizaje automático (Machine Learning) SINDy. Este algoritmo construye ecuaciones dinámicas a partir de series temporales. Su implementación se llevó a cabo en Python. En la primera parte del presente estudio:

- Se elaboraron con éxito códigos utilizando Python, el lenguaje de programación que ha sido seleccionado para abordar el problema planteado en esta tesis. Específicamente, se desarrollaron códigos en Python para calcular funciones objetivo/ajuste mediante regresión lineal con mínimos cuadrados (que son la base de SINDy y, en general, de Machine Learning). Usando bases de datos disponibles en línea, se compararon los resultados obtenidos con estos códigos desarrollados (Cuadro 5.1) con aquellos que arrojan las librerías ya existentes (Scikitlearn) mostrando un excelente acuerdo entre ambos procedimientos. Esto a manera de práctica para aprender y mejorar las habilidades en el manejo del lenguaje de programación Python.
- Usando la paquetería PySINDy, se reprodujeron correctamente las ecuaciones dinámicas de los sistemas analizados en el anexo del artículo de Steven Brunton [16] y en donde únicamente se muestran resultados en forma gráfica. En esta Tesis se calcula y reporta cuantitativamente la precisión del modelo reconstruido. Las series temporales del modelo reconstruido presentan un error cuadrático medio (ϵ_{MSE}) del orden de $\approx 10^{-10}$ y un coeficiente de correlación aproximado de ≈ 1 respecto de las trayectorias del modelo original. Estos resultados destacan la

eficacia de la paquetería PySINDy para la (re)construcción correcta de las ecuaciones dinámicas en sistemas sencillos, como el oscilador armónico y el péndulo simple.

- En el caso particular del atractor de Lorenz y a diferencia del oscilador armónico o el péndulo simple (sistemas integrables), al reconstruir las ecuaciones dinámicas con PySINDy, se observa una eventual pérdida significativa en las predicciones del modelo reconstruido. Esto justifica el estudio de la precisión del algoritmo de aprendizaje automático en la (re)construcción de modelos que describen sistemas caóticos.

Después se propuso una generalización del célebre sistema Hamiltoniano de Hénon-Heiles para estudiar el algoritmo SINDy en presencia de caos. Se realizó un análisis sistemático de la dinámica como función del parámetro N que aparece en el potencial del sistema y la energía:

- Se construyeron las secciones de superficie de Poincaré en los casos $N = 3$ (potencial polinomial cúbico) y $N = 4$ (potencial polinomial cuártico). Usando las líneas de simetría se obtuvieron condiciones iniciales de órbitas periódicas.
- En general, para este sistema, el algoritmo SINDy produce resultados aproximados donde para las trayectorias se obtiene un error cuadrático medio $\epsilon_{MSE} \approx 10^{-10}$ y coeficiente de correlación $R^2 \approx 1$, indicando la precisión del algoritmo independientemente del tipo de trayectoria y energía del sistema. Se define el límite dt_c para el cual los modelos obtenidos dejan de ser parsimonios (sencillos y precisos), debido a una degradación del error cuadrático medio y el coeficiente de correlación por la aparición de términos extras.
- Se señala en el caso $N = 4$, que cuando las series de tiempo admiten una expansión finita en la base de funciones candidatas, el algoritmo de aprendizaje automático no es capaz de reproducir las ecuaciones de movimiento de forma correcta. Por lo cual la reconstrucción debe considerar mas de una serie temporal para encontrar las ecuaciones dinámicas esperadas.
- Es notable que en situaciones donde se conoce de forma exacta el modelo y para las cuales no se precisa de la reconstrucción de las ecuaciones gobernantes, SINDy

puede ser también aprovechado para derivar expresiones analíticas aproximadas de algunas trayectorias periódicas del sistema, y posiblemente de cantidades conservadas.

Por último, como un complemento adicional se estudió de forma introductoria la reconstrucción de las ecuaciones gobernantes para un modelo que genera señales de electrocardiogramas. En este modelo se presenta una transición entre dinámicas periódicas y caóticas. Específicamente:

- El algoritmo SINDy reconstruye correctamente las ecuaciones dinámicas del modelo de electrocardiogramas, para las trayectorias estudiadas, usando valores del orden 10^{-4} en el paso de integración dt , tal y como sucedía en la generalización del sistema de Hénon-Heiles. En el caso de órbitas caóticas se aprecia una disminución en la precisión del modelo ($R^2 = 0,51115$) en comparación con las órbitas (no-caóticas) donde $R^2 \approx 1$.
- Finalmente, se planteó la construcción de las ecuaciones gobernantes en un sistema caótico donde únicamente se conoce de forma parcial la dinámica (no se tienen las mediciones de todas las variables). Se reconstruye las series temporales faltantes mediante el teorema de Takens. Con base en los resultados obtenidos al aplicar este procedimiento al atractor de Lorenz, se sugiere su exploración en el modelo de electrocardiogramas como una dirección para futuras investigaciones.

Bibliografía

- [1] Henri Poincaré. The value of science: essential writings of Henri Poincaré. Modern library, 2012.
- [2] Michael Marsalli. Mcculloch-pitts neurons. 1161:1162, 2006.
- [3] Michael Tabor. Chaos and integrability in nonlinear dynamics: An introduction, wileyinterscience. Chaos and Integrability in Nonlinear Dynamics: An Introduction, 1989.
- [4] EJ Moulton. H. goldstein, classical mechanics. 1952.
- [5] Melvin G Calkin. Lagrangian and Hamiltonian mechanics. World Scientific, 1996.
- [6] E Piña and L Jiménez Lara. On the symmetry lines of the standard mapping. Physica D: Nonlinear Phenomena, 26(1-3):369–378, 1987.
- [7] A.M. Escobar-Ruiz, L. Jiménez-Lara, P.M. Juárez-Flores, F. Montoya-Molina, J. Moreno-Sáenz, and M.A. Quiroz-Juárez. Data-driven reconstruction of chaotic dynamical equations: The hénon–heiles type system. Chaos, Solitons Fractals, 184: 115025, 2024. ISSN 0960-0779. doi: <https://doi.org/10.1016/j.chaos.2024.115025>.
- [8] R DeVogelaere. Contributions to the theory of nonlinear oscillations. Vol. IV, page 53, 1958.
- [9] Wilhelm Kutta. Beitrag zur näherungsweisen Integration totaler Differentialgleichungen. Teubner, 1901.
- [10] Geeta Arora, Varun Joshi, and Isa Sani Garki. Developments in runge–kutta method to solve ordinary differential equations. In Recent Advances in Mathematics for Engineering, pages 193–202. CRC Press, 2020.

- [11] Leonhard Euler. INSTITVTIONVM CALCVLI INTEGRALIS VOLVMEN PRIMVM IN QVO METHODVS INTEGRANDI A PRIMIS PRINCIPIIS VSQVE AD INTEGRATIONEM AEQVATIONVM DIFFERENTIALIVM PRIMI GRADVS PERTRACTATVR, volume 1. Impensis Academiae Imperialis Scientiarum, 1768.
- [12] Erwin Fehlberg. Classical fourth-and lower order runge-kutta formulas with step-size control and their application to heat transfer problems. Computing, 6:61–71, 1970.
- [13] Kees Aarts. Parsimonious methodology. Methodological Innovations Online, 2(1): 2–10, 2007.
- [14] Robert Tibshirani. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society Series B: Statistical Methodology, 58(1):267–288, 1996.
- [15] AK Md Ehsanes Saleh, Mohammad Arashi, and BM Golam Kibria. Theory of ridge regression estimation with applications. John Wiley & Sons, 2019.
- [16] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proceedings of the national academy of sciences, 113(15):3932–3937, 2016.
- [17] Oliver Nelles and Oliver Nelles. Nonlinear dynamic system identification. Springer, 2020.
- [18] SL Brunton and JN Kutz. Data driven science & engineering-machine learning, dynamical systems, and control (p. 572). databook. uw. edu. 2017.
- [19] Ignacio Arnaldo, Una-May O’Reilly, and Kalyan Veeramachaneni. Building predictive models via feature synthesis. In Proceedings of the 2015 annual conference on genetic and evolutionary computation, pages 983–990, 2015.
- [20] Joshua L Proctor, Steven L Brunton, Bingni W Brunton, and JN Kutz. Exploiting sparsity and equation-free architectures in complex systems. The European Physical Journal Special Topics, 223(13):2665–2684, 2014.
- [21] Alan A Kaptanoglu, Brian M de Silva, Urban Fasel, Kadierdan Kaheman, Andy J Goldschmidt, Jared L Callahan, Charles B Delahunt, Zachary G Nicolaou, Kathleen Champion, Jean-Christophe Loiseau, et al. Pysindy: A comprehensive python package for robust sparse system identification. arXiv preprint arXiv:2111.08481, 2021.
- [22] Kaushal Kumar. Data-driven modeling and parameter estimation of nonlinear

- systems. The European Physical Journal B, 96(7):107, 2023.
- [23] Edward N Lorenz. Deterministic nonperiodic flow. Journal of atmospheric sciences, 20(2):130–141, 1963.
 - [24] Michel Henon and Carl Heiles. The applicability of the third integral of motion: Some numerical experiments. Astronomical Journal, 69:73, February 1964. doi: 10.1086/109234.
 - [25] M Álvarez-Ramírez and JD García-Saldaña. Periodic orbits of a generalized hénon–heiles system. Journal of Physics A: Mathematical and Theoretical, 53(6):065204, 2020.
 - [26] Sawsan Alhowaity, Elbaz I Abouelmagd, Zouhair Diab, and Juan LG Guirao. Calculating periodic orbits of the hénon–heiles system. Frontiers in Astronomy and Space Sciences, 9:945236, 2023.
 - [27] Jaume Llibre and Lidia Jiménez-Lara. Periodic orbits and non-integrability of hénon–heiles systems. Journal of Physics A: Mathematical and Theoretical, 44(20):205103, 2011.
 - [28] Willard Miller, Sarah Post, and Pavel Winternitz. Classical and quantum superintegrability with applications. Journal of Physics A: Mathematical and Theoretical, 46(42):423001, 2013.
 - [29] AM Escobar-Ruiz, R Linares, and P Winternitz. New infinite families of nth-order superintegrable systems separating in cartesian coordinates. Journal of Physics A: Mathematical and Theoretical, 53(44):445203, 2020.
 - [30] James Keener and James Sneyd. Mathematical physiology: II: Systems physiology. Springer, 2009.
 - [31] Yi Zhao, Junfeng Sun, and Michael Small. Evidence consistent with deterministic chaos in human cardiac data: surrogate and nonlinear dynamical modeling. International Journal of Bifurcation and Chaos, 18(01):141–160, 2008.
 - [32] MA Quiroz-Juárez, O Jiménez-Ramírez, R Vázquez-Medina, V Breña-Medina, JL Aragón, and RA Barrio. Generation of ecg signals from a reaction-diffusion model spatially discretized. Scientific reports, 9(1):19000, 2019.
 - [33] Omer Berenfeld and Shimon Abboud. Simulation of cardiac activity and the ecg using a heart model with a reaction-diffusion action potential. Medical engineering & physics, 18(8):615–625, 1996.
 - [34] Mark Potse, Bruno Dubé, Jacques Richer, Alain Vinet, and Ramesh M Gulrajani.

- A comparison of monodomain and bidomain reaction-diffusion models for action potential propagation in the human heart. IEEE Transactions on Biomedical Engineering, 53(12):2425–2435, 2006.
- [35] Alan Garfinkel, Peng-Sheng Chen, Donald O Walter, Hrayr S Karagueuzian, Boris Kogan, Steven J Evans, Mikhail Karpoukhin, Chun Hwang, Takumi Uchida, Masamichi Gotoh, et al. Quasiperiodicity and chaos in cardiac fibrillation. The Journal of clinical investigation, 99(2):305–314, 1997.
- [36] Floris Takens. Detecting strange attractors in turbulence. In Dynamical Systems and Turbulence, Warwick 1980: proceedings of a symposium held at the University of Warwick 1979/80, pages 366–381. Springer, 2006.
- [37] Henry Abarbanel. Analysis of observed chaotic data. Springer Science & Business Media, 2012.
- [38] Jiri Horak, Ladislav Krlín, and Ales Raidl. Deterministický chaos a jeho fyzikální aplikace. Academia, 2003.



Apéndice A

A.1. Código en Python: Integración numérica

A continuación, se exponen los códigos implementados en el lenguaje de programación Python, diseñados por el autor de este trabajo, con el propósito de abordar la resolución de una ecuación diferencial mediante dos enfoques: el método de Euler y el método de Runge-Kutta de orden cuatro. Estas implementaciones se fundamentan en el concepto de integración numérica para obtener soluciones aproximadas de la ecuación diferencial dada.

```
####Importar Librerias
import numpy as np
import matplotlib.pyplot as plt
#####
### Ecuación Diferencial
def f(t,x):
    return (np.power(t,2))*(np.sin(t))
#####
#Datos
N=501
ti=1
tf=60
delta_t=(tf-ti)/(N) #tamaño del número de pasos
print(delta_t)
```

```

t=np.linspace(ti,tf,N) # aqui estan las entradas para los valores de x
x=np.zeros([nd]) #se usa un "array" (lista ordenada con una columna o fila)
#####
print('Mtodo de Euler')
#Se calcula cada punto x_n
x[0]=ti #condicin inicial
for k in range(1,N):
    x[k]= f(t[k-1],x)
#Imprimir resultados
print('t[k]\tx[k]')
print('-----')
#####
#Graficamos los resultados usando matplotlib
# solucin aproximada:
plt.plot(t,x,'o')
# Solucin exacta:
plt.plot(t, -((-2+np.power(t,2))*np.cos(t))+2*xd*np.sin(t)
, 'g', label='Exact')
plt.xlabel("t")
plt.ylabel(" x(t)")
plt.title("Comparacin del Mtodo de Euler vs Solucin Exacta")
plt.show()
#####
#####
#####
print('Mtodo RK-4')
dt=np.divide((tf-ti),N) #calculo del tamao de pasos
t=np.linspace(ti,tf,N) #tabla para valores de x
x_rk=np.zeros([N]) #aqui se almacenan los resultados de y
print('Paso de integracin dt=',dt)
print('-----')
print(' x[i] | y[i] | ')
print('-----|-----|-----')
for i in range(N-1):#se establece N1-1 para arreglar el conteo
    x_rk[0]=ti #Se define la condicion inicial
    print(x_rk[0])

```



```

##### Mtodo RK-4
K1=f(t[i],x_rk[i])#k1
K2=f(t[i]+np.divide(dt,2),x_rk[i]+K1*np.divide(dt,2))#k2
K3=f(t[i]+np.divide(dt,2),x_rk[i]+K2*np.divide(dt,2))#k3
K4=f(t[i]+dt,x_rk[i]+K3*dt)#k4
x_rk[i+1]=x_rk[i]+np.divide(dt,6)*(K1+2*K2+2*K3+K4)
#Imprimir los valores de cada iteracin:
print('%f%f%f%f%f' % (x_rk[i],t[i]))

#####
###Comparacin entre mtodos numricos
# Solucin con RK4
plt.plot(t,x_rk,'.',label='RK-4',color='red')
# Solucin Exacta
plt.plot(t, -((-2+np.power(t,2))*np.cos(t))+2*t*np.sin(t)
, '- ',color='black',label='Exacto')
# solucin con mtodo de Euler
plt.plot(t,x,':',color='blue',label='Euler')
plt.legend(loc='best')
plt.xlabel("t",fontsize=17)
plt.ylabel("x(t)",fontsize=17)

```


B.1. Demostración para la solución multivariable de mínimos cuadrados

A continuación se realiza la demostración de la solución (5.34), usada para encontrar los pesos en el problema de mínimos cuadrados del caso lineal multivariable, correspondientes a la ecuación (5.24),

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k , \quad (\text{B.1})$$

con $\epsilon = \mathbf{Y} - \mathbf{XB}$.

Se resuelve así, el problema de minimización para el error cuadrático medio (5.31) de la forma,

$$\frac{\partial(\epsilon^T \epsilon)}{\partial \mathbf{B}} = 0 ; \quad (\text{B.2})$$

primero se escribe $\epsilon^T \epsilon$ de la siguiente forma:

$$\begin{aligned} \epsilon^T \epsilon &= [\mathbf{Y} - \mathbf{XB}]^T [\mathbf{Y} - \mathbf{XB}] \\ &= [\mathbf{Y}^T - (\mathbf{XB})^T] [\mathbf{Y} - \mathbf{XB}] \\ &= [\mathbf{Y}^T - \mathbf{B}^T \mathbf{X}^T] [\mathbf{Y} - \mathbf{XB}] \\ &= \mathbf{Y}^T \mathbf{Y} - \mathbf{Y}^T \mathbf{XB} - \mathbf{B}^T \mathbf{X}^T \mathbf{Y} + \mathbf{B}^T \mathbf{X}^T \mathbf{XB} \end{aligned}$$

$$\epsilon^T \epsilon = \mathbf{Y}^T \mathbf{Y} - 2\mathbf{Y}^T \mathbf{X} \mathbf{B} + \mathbf{B}^T \mathbf{X}^T \mathbf{X} \mathbf{B} \quad (\text{B.3})$$

Tomando la derivada en B.3, igualando a cero y resolviendo para $\mathbf{B}$ se obtiene lo siguiente,

$$\frac{\partial(\epsilon^T \epsilon)}{\partial \mathbf{B}} = -2\mathbf{X}^T \mathbf{Y} + 2\mathbf{X}^T \mathbf{X} \mathbf{B} = 0 \quad (\text{B.4})$$

$$(\mathbf{X}^T \mathbf{X}) \mathbf{B} = \mathbf{X}^T \mathbf{Y} \quad (\text{B.5})$$

multiplicando esta última expresión por la inversa de $\mathbf{X}^T \mathbf{X}$,

$$\underbrace{(\mathbf{X}^T \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{X})}_{\mathbf{I}} \mathbf{B} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}, \quad (\text{B.6})$$

en donde el término $(\mathbf{X}^T \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{X})$ es igual a la matriz identidad $\mathbf{I}$; este producto tendrá inversa siempre que $\mathbf{X}$ sea sobredeterminado. Por lo que se concluye la siguiente expresión para calcular la matriz $\mathbf{B}$ de los coeficientes β_k que resuelven el problema de regresión

$$\mathbf{B} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}. \quad (\text{B.7})$$

B.2. Código en Python: Regresión lineal univariable

Se presenta el código de regresión lineal que encuentra los pesos β_0, β_1 de la función objetivo:

$$y = \beta_0 + \beta_1 x_1,$$

en este caso del ejemplo en la Sección 5.2.2, para calcular la esperanza de vida de una persona a partir del índice de masa corporal.

```
####Importar librerias
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.metrics import mean_squared_error, r2_score
```

```
#####
####Lectura de datos
bmi_life_data= pd.read_csv('bmi_and_life_expectancy.csv') #leer los datos
x = bmi_life_data[['BMI']] #datos en x
arr=np.array(x) #arreglo de datos
X=np.concatenate(arr)
y =bmi_life_data[['Life expectancy']] #datos en y
arr1=np.array(y) #arreglo de datos
Y=np.concatenate(arr1)
#####
###Calcular de promedios
xprom=np.sum(X)/len(X) #promedio en x
yprom=np.round(np.sum(Y)/len(X),3) #promedio en y
#print('xprom=',xprom)
#print('yprom=',yprom)
#####
###Calcular de pesos
b1= np.sum(X*(Y-yprom))/np.sum(X*(X-xprom))#beta_{1}
print('beta_1=',round(b1,2))
b0= yprom-b1*xprom #beta_{0}
print('beta_0=',round(b0,2))
def f(x1): #funcion de ajuste lineal
    return b1*x1+b0
#####
###Error cuadrático medio
mse_tab=[]
for i in range(len(X)):
    mse_tab.append(((Y[i]-f(X[i]))**2))
mse=(1/len(Y))*np.sum(mse_tab)
print('MSE=',mse)
#####
#Grafica de datos originales
plt.plot(X, Y, '.',color='black')
#Grafica de la linea de ajuste
plt.plot(X,b1*X+b0,color='red',label='$\hat{y}=b_0+b_1 x$')
```

```
plt.xlabel('x',fontsize=17)
plt.ylabel('y',fontsize=17)
plt.legend(loc='lower right')
plt.show()
```

```
beta_1= 2.52
beta_0= 7.23
MSE= 43.02103167161064
```

B.3. Programa en Python: Regresión lineal multi-variable

Se presenta el código para el caso de la regresión lineal para dos variables independientes donde se encuentran los pesos de una función objetivo tipo:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

```
####Importar. librerias
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
#####
####Lectura de Datos
data1 = pd.read_csv('Real estate.csv')
X      = data1[['X2 house age','X4 number of convenience stores']]
Y      = data1[['Y house price of unit area']]
####Arreglo de la matriz de datos
#Se agrega una columna de unos a la matriz X
unos = np.ones((len(X), 1))#vector columna de unos
X = pd.concat([pd.DataFrame(unos), X], axis=1)#Arreglo de X
#print(X)
#####
x1 = data1[['X2 house age']] #datos en x1
arr1=np.array(x1) #arreglo de datos x1
```

```

X1=np.concatenate(arr1)
x2 = data1[['X4 number of convenience stores']] #datos en x2
arr2=np.array(x2) #arreglo de datos x2
X2=np.concatenate(arr2)
XX=np.array(list(zip(X1,X2)))
y = data1[['Y house price of unit area']] #datos de y
arr_y=np.array(y) #arreglo de datos "y"
yy=np.concatenate(arr_y)
#print(XX)
#####
####Calculo de matriz B=(XT*X)-1*XT*Y
x=X.T #Transpuesta de X
w=np.dot(x,X) #Producto XT*X
inv1=np.linalg.inv(w) #inversa de (XT*X)
#comment:existe una inversa solo si X es sobredeterminado
w1=np.dot(inv1,x) #Producto (XT*X)-1*XT
betas=np.dot(w1,Y) #Resultado para los pesos en B
####Resultados
print('B=',betas)
print('Funcin de ajuste: y=',betas[0],'+',betas[1], 'x1', '+',betas[2], 'x2')
#####
####Error cuadrático medio
def f(x1,x2): #funcion de ajuste lineal
    return betas[1]*x1+betas[2]*x2+betas[0]
mse_tab=[] #Lista de error del i-simo dato
for i in range(len(X)):
    mse_tab.append(((yy[i]-f(X1[i],X2[i]))**2))
mse=(1/len(yy))*np.sum(mse_tab)#calculo de error total
print('MSE=',mse)
####Coeficiente de correlación
yhat=[] #Datos obtenidos por la funcin de ajuste
for j in range(len(yy)):
    yhat.append(f(X1[j],X2[j]))
yprom=np.sum(yy)/len(yy) #promedio en y
R2n=np.sum(np.power(yhat-yprom,2))#numerador del coef. de correlación
R2d=np.sum(np.power(yy-yprom,2))#denominador del coef. de correlación

```

```
r2=R2n/R2d #coeficiente de correlacin  
print('R2=',r2)
```


C.1. Código en Python: Regularización STLQ

En este Apéndice se muestra el código para la ilustración de la técnica de regularización STLQ, presentada en la Sección 5.3, sobre el ejemplo de la obtención de un modelo disperso para una base de datos aleatoria.

```
#####Ejemplo Caso Multivariable#####
#####
#### Importar librerías necesarias
import numpy as np
#####
####Lectura de Datos
#Matriz de datos para un conjunto de cuatro variables independientes
X=np.array([[1,76,80,13.5],[1,61,72,12.1],[1,50,70,11.6],[1,94,122,12.5],
            [1,55,75,13.5],[1,61,95,14],[1,80,120,12.5],[1,52,68,14.5]])
#Matriz de datos para la variable dependiente
Y=np.array([[250],[220],[200],[350],[210],[205],[285],[190]])
# Definir tamaño de la matriz de datos X
n = 3 #numero de variables
m = 8 #numero de datos
print('X=\n',X)
#####
#### Cálculo de solución dispersa
lambda_val = 1.5 #valor aleatorio del umbral
```

```

# Calcular una estimacin inicial de los
coeficientes de regresin usando mnimos cuadrados
Xi = np.linalg.lstsq(X, Y, rcond=None)[0]
print('Pesos del primer calculo de regresin(LS):\n',Xi)
num_iterations = 3 # Nmero de iteraciones para STLQ
#### Algoritmo STLQ
# Iterar para obtener una solucin dispersa
para los coeficientes de regresin
for i in range(num_iterations):
    # Seleccionar los coeficientes con magnitud menor que el umbral
    small_inds = np.abs(Xi) < lambda_val
    Xi[small_inds] = 0
    #print(small_inds)
    print('Xi para la iteracin',i+1,'\n',Xi)
    # Regresar dinmicas a travs de los trminos restantes
    para encontrar Xi escaso
    for j in range(n-3): #el rango depende de (m,n) en dXdt
        big_inds = np.logical_not(small_inds[:, j])#para tomar
        en cuenta los valores mayores a lambda
        #print('valores que son >lambda\n',big_inds)
        Xi[big_inds, j] = np.linalg.lstsq(X[:, big_inds]
        , dXdt.iloc[:, j], rcond=None)[0]
print('Resultado de Xi disperso con',num_iterations
,'iteraciones:\n', Xi)# Imprimir resultados

# Calcular las predicciones
Y_pred = np.dot(X, Xi)

# Calcular el Error Cuadrático Medio (MSE)
MSE = np.mean((dXdt - Y_pred)**2)

# Calcular el coeficiente de correlacin (R cuadrado)
corr_coef = np.corrcoef(dXdt.T, Y_pred.T)[0, 1]**2

```

```
print('Error Cuadrático Medio (MSE):', MSE)
print('Coeficiente de correlación (R cuadrado):', corr_coef)

#Resultados
X=
[[ 1.  76.  80.  13.5]
 [ 1.  61.  72.  12.1]
 [ 1.  50.  70.  11.6]
 [ 1.  94. 122.  12.5]
 [ 1.  55.  75.  13.5]
 [ 1.  61.  95.  14. ]
 [ 1.  80. 120.  12.5]
 [ 1.  52.  68.  14.5]]
Pesos del primer calculo de regresión(LS):
[[121.70448779]
 [ 2.94932354]
 [ 0.27556962]
 [-7.84335821]]
Xi para la iteración 1
[[121.70448779]
 [ 2.94932354]
 [ 0.          ]
 [-7.84335821]]
Xi para la iteración 2
[[121.70448779]
 [ 2.94932354]
 [ 0.          ]
 [-7.84335821]]
Xi para la iteración 3
[[121.70448779]
 [ 2.94932354]
 [ 0.          ]
 [-7.84335821]]
Resultado de Xi disperso con 3 iteraciones:
[[121.70448779]
 [ 2.94932354]
```

```
[ 0.      ]
[ -7.84335821]]
Error Cuadrático Medio (MSE): 728.2657566096652
Coeficiente de correlación (R cuadrado): 0.9539206536933651
```

C.2. Código en Python: Frente de Pareto

A continuación, se proporciona la implementación en Python de la técnica frente de Pareto presentada en 5.4, para un conjunto de datos aleatorios. En este contexto, se obtiene un modelo disperso mediante la selección de un valor del hiperparámetro λ , el cual regula la regularización, utilizando el método LASSO.

```
# Importar librerías necesarias
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
# Definir entradas de ejemplo
n = 4 #numero de variables
m = 8 #numero de datos
# Generar datos de entrada aleatorios
Theta = np.array([[1,76,80,13.5],[1,61,72,12.1],[1,50,70,11.6],
                  ,[1,94,122,12.5],[1,55,75,13.5],[1,61,95,14],
                  ,[1,80,120,12.5],[1,52,68,14.5]])
dXdt = np.array([[250],[220],[200],[350],[210],[205],[285],[190]])
print('Theta=\n',Theta)
#####Algoritmo de dispersin
def LASSO(Theta,dXdt,lambdaval): #no es lasso, es el algoritmo del paper
    Xi = np.linalg.lstsq(Theta, dXdt, rcond=None)[0]
    #print('Xi_primer calculo LS:\n',Xi)
    # Definir el nmero de iteraciones
    num_iterations = 7 # EN ESTE CASO DE CV DEJAR ESTO EN 1
    # Iterar para obtener una solución escasa para los coeficientes de regresión
    for i in range(num_iterations):
        # Seleccionar los coeficientes con magnitud menor que el umbral
        small_inds = np.abs(Xi) < lambdaval
        Xi[small_inds] = 0
```

```
#print(small_inds)
print('Xi para la iteracin',i+1,'\n',Xi)
# Regresar dinmicas a travs de los trminos
restantes para encontrar Xi escaso
for j in range(n-3): #el rango depende de (m,n) en dXdt
    big_inds = np.logical_not(small_inds[:, j])#para tomar en cuenta
    los valores mayores a lambda
    #print('valores que son >lambda\n',big_inds)
    Xi[big_inds, j] = np.linalg.lstsq(X[:, big_inds]
    , dXdt.iloc[:, j], rcond=None)[0]
# Imprimir resultados
print('Resultado de Xi disperso con',num_iterations,'iteraciones:\n', Xi)
```


D.1. Cálculo de condiciones iniciales aleatorias que conservan energía

Para construir la sección de superficie de Poincaré en el caso $N = 3$ se requiere un conjunto de condiciones iniciales aleatorias en donde se considere el corte del espacio fase en el plano con $x = 0$ (también podría considerarse un corte con $y = 0$). Estas condiciones deben respetar la condición de conservación de energía dado de la forma

$$E \equiv E(x, p_x, y, p_y) = \frac{p_x^2 + p_y^2}{2} + \frac{x^2 + y^2}{2} + x^2 y - \frac{y^3}{3}, \quad (\text{D.1})$$

considerando que $x = 0$, la condición anterior se reescribe como

$$E(x = 0, p_x, y, p_y) = \frac{p_x^2 + p_y^2}{2} + \frac{y^2}{2} - \frac{y^3}{3}. \quad (\text{D.2})$$

De esta ecuación se construyen las condiciones que deben cumplir los valores y , p_y , p_x . Primero se encuentra una expresión analítica para el valor del momento p_x a partir de (D.2), que es de la forma

$$p_x = \sqrt{\underbrace{2E - y^2 + \frac{2y^3}{3} - p_y^2}_S}, \quad (\text{D.3})$$

para obtener valores reales de p_x el término dentro del radical, llamándolo S , debe ser mayor o igual a cero ($S \geq 0$); en el caso de $S = 0$ tendremos soluciones que se encuentran sobre de la región acotada en la sección de superficie de Poincaré y definen el contorno en el cual viven las soluciones para $S > 0$. Se considera entonces, para encontrar el contorno de la región acotada, la ecuación:

$$2E - y^2 + \frac{2y^3}{3} - p_y^2 = S = 0 , \quad (\text{D.4})$$

para la que se puede encontrar una expresión de los valores que debe tomar p_y sobre el contorno. Resolviendo, $S = 0$ en (D.4), se obtiene

$$p_y = \sqrt{2E - y^2 + \frac{2y^3}{3}} . \quad (\text{D.5})$$

Por lo tanto se pueden dar valores aleatorios para y y así calcular su valor correspondiente de p_x y p_y con las ecuaciones (D.3),(D.5), donde por construcción, todas las condiciones iniciales que se generen cumplirán con la conservación de la energía, dichas condiciones pueden generar trayectorias periódicas (las que se encuentran en los puntos fijos), cuasiperiódicas (las que orbitan alrededor de los puntos elípticos o hiperbólicos) o incluso trayectorias caóticas.

Pero se necesita definir los intervalos de la región donde se conserva la energía, la cual se define en el intervalo de los valores y_{min} , y_{max} , los cuales se obtienen considerando $p_y = 0$. Entonces, se resuelve:

$$p_y = \sqrt{2E - y^2 + \frac{2y^3}{3}} = 0 \quad (\text{D.6})$$

Resolviendo con ayuda de *Mathematica* la ecuación cubica que sale de la ecuación anterior, se obtienen los siguientes resultados que únicamente dependen del valor de energía que se este trabajando,

$$y_1 \rightarrow \frac{1}{2} \left(\sqrt[3]{-12E + 2\sqrt{6}\sqrt{E(6E-1)} + 1} + \frac{1}{\sqrt[3]{-12E + 2\sqrt{6}\sqrt{E(6E-1)} + 1}} + 1 \right) , \quad (\text{D.7})$$

$$y_2 \rightarrow \frac{1}{4} \left(i \left(\sqrt{3} + i \right) \sqrt[3]{-12E + 2\sqrt{6}\sqrt{E(6E-1)} + 1} - \frac{1 + i\sqrt{3}}{\sqrt[3]{-12E + 2\sqrt{6}\sqrt{E(6E-1)} + 1}} + 2 \right), \quad (\text{D.8})$$

$$y_3 \rightarrow \frac{1}{4} \left(2 - \left((1 + i\sqrt{3}) \sqrt[3]{-12E + 2\sqrt{6}\sqrt{E(6E-1)} + 1} \right) + \frac{i(\sqrt{3} + i)}{\sqrt[3]{-12E + 2\sqrt{6}\sqrt{E(6E-1)} + 1}} \right). \quad (\text{D.9})$$

Por ejemplo, al asignar el valor de la energía $E = 0,125$, se encuentra que el y_{min} , y_{max} corresponden a los valores siguientes,

$$y_{min} = y_2 = -0,439693 \quad (\text{D.10})$$

$$y_{max} = y_3 = 0,673648 \quad (\text{D.11})$$

En el caso de $y_1 = 1,26604$, si se usa para generar las condiciones iniciales, se obtienen valores imaginarios de p_x , es decir el término dentro del radical queda negativo, opuesto a lo que se estaba buscando. Este último calculo se realiza para los diferentes valores de energía de los cuales se busca construir la sección de superficie de Poincaré.

Para construir las secciones de superficie del caso $N = 4$, se sigue un procedimiento como el anterior, de donde se obtiene lo siguiente. Considerando ahora

$$E \equiv E(x, p_x, y, p_y) = \frac{p_x^2 + p_y^2}{2} + \frac{x^3 y - y^3 x}{2}, \quad (\text{D.12})$$

considerando el plano en $x = 0$, la expresión anterior se escribe como

$$E = \frac{1}{2}(p_x^2 + p_y^2) + \frac{1}{2}y^2, \quad (\text{D.13})$$

resolviendo para p_x ,

$$p_x = \sqrt{\underbrace{2E - y^2 - p_y^2}_S}. \quad (\text{D.14})$$

Para valores sobre el contorno de la región acotada, donde $S = 0$, se resuelve para p_y

en la ecuación (D.14) y se obtiene,

$$2E - y^2 - p_y^2 = 0 \quad (\text{D.15})$$

$$p_y = \sqrt{2E - y^2} , \quad (\text{D.16})$$

de donde se encuentra, mediante la expresión cuadrática en el radical de la ecuación (D.16), los intervalos de la región donde se pueden encontrar condiciones iniciales que conservan energía:

$$\begin{aligned} y_{max} &= \sqrt{2E} \\ y_{min} &= -\sqrt{2E} . \end{aligned} \quad (\text{D.17})$$

E.1. Código en Python: Reconstrucción de ecuaciones con SINDy

A continuación se muestra un ejemplo del tipo de código que se usó para este trabajo, usando la paquetería PySINDy junto con una técnica de regularización (STLQ) y de optimización (frente de Pareto).

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.cm import rainbow
from scipy.integrate import solve_ivp
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score
import pysindy as ps

# ignorar advertencias
import warnings
warnings.filterwarnings("ignore", category=UserWarning)
np.random.seed(1000)

# Integrator keywords for solve_ivp
integrator_keywords = {}
integrator_keywords['rtol'] = 1e-12
integrator_keywords['method'] = 'RK45'
```

```

integrator_keywords['atol'] = 1e-12
#####
# Make coefficient plot for threshold scan
def plot_pareto(coefs, opt, model, threshold_scan, x_test, t_test):
    dt = t_test[1] - t_test[0]
    mse = np.zeros(len(threshold_scan))

    mse_sim = np.zeros(len(threshold_scan))
    #####
    for i in range(len(threshold_scan)):
        opt.coef_ = coefs[i]
        mse[i] = model.score(x_test, t=dt,
                             metric=mean_squared_error)
        #Extrapolation?
        ##
        x_test_sim = model.simulate(x_test[0, :], t_test, integrator="odeint")
        #interpolation?
        if np.any(x_test_sim > 1e4):
            x_test_sim = 1e4
        mse_sim[i] = np.sum((x_test - x_test_sim) ** 2)
    #####
    plt.figure()
    plt.semilogy(threshold_scan, mse, "bo")
    plt.semilogy(threshold_scan, mse, "b")
    plt.ylabel(r"MSE", fontsize=20)
    plt.xlabel(r"$\lambda$", fontsize=20)
    plt.xticks(fontsize=16)
    plt.yticks(fontsize=16)
    plt.grid(True)
    plt.title('Interpolacin')
    plt.figure()
    plt.semilogy(threshold_scan, mse_sim, "bo")
    plt.semilogy(threshold_scan, mse_sim, "b")
    plt.ylabel(r"MSE", fontsize=20)
    plt.xlabel(r"$\lambda$", fontsize=20)
    plt.xticks(fontsize=16)

```

```

plt.yticks(fontsize=16)
plt.grid(True)
plt.title('Extrapolacin')
plt.show()
#####
#Ecuaciones de Hamilton
def HH(y, t):
    q1,q2,dq1,dq2 = y # Desempaquetamos los valores de y
    dydt = [dq1,dq2,-q1-2*q1*q2,-q2-np.power(q1,2)+np.power(q2,2)]
    return dydt
# Generate training data
feature_names=['x','y','px','py']
dt = 0.0001 #paso de separacion entre los datos de entrenamiento
t_train = np.arange(0, 250, dt)# lista temporal de entrenamiento
t_train_span = (t_train[0], t_train[-1])
#print(t_train)
#print(len(t_train))
#print(t_train_span)
#####
x0_per14 = [0,-0.2134150807894881,0.176561230871,0]
x0_cua14 = [0,0.1,0.271784,0]
#####
x0_per34 = [0,0.3038006035414,0.41999762532241397,0] #Periodica 3_4
x0_cua34 = [0,0.15,0.479322,0] #cuasi periodica 3_4
x0_cao34 = [0,0.25,0.328507,-0.3] #catica 3_4
#####
x0_per166 = [0,0.5473794640724625,0.37821665749647776,0]
x0_cua166 = [0,0.6,0.342345,0]
x0_cao166 = [0,0.1,0.520545,0.23]
#####
#Se genera una lista de datos al reconstruir el sistema
#x_train = odeint(HH,x0_cua, t_train)
#print(x_train)
#print(len(x_train))
#plt.plot(x_train[:,0],x_train[:,1])
### Se ajustan los datos de entrenamiento en un modelo de SINDy

```

```

#####orden del polinomio
poly_order = 3
#####threshold para los valores de los coeficientes
threshold = 0.05
##### Librerias candidatas
polynomial_library=ps.PolynomialLibrary(degree=poly_order)
feature_names=['x','y','px','py']
#####
#Se define la condicin inicial para
#el entrenamiento y validacin
x0=x0_per166
x0t=x0_cao166
#####
#paso de integracin
dt1=0.001
print('dt_train:',dt1)
#intervalo de tiempo
t_train1 = np.arange(0, 300, dt1)
#Generacin de datos para entrenar a SINDy
x_train1 = odeint(HH,x0 , t_train1)
#Generacin de datos para validar a SINDy
x_test1 = odeint(HH,x0t , t_train1)#####
#Datos exactos
x_exact = odeint(HH,x0 , t_train )#####
#Ejste de SINDy
model1 = ps.SINDy(
    feature_names=feature_names,
    feature_library=polynomial_library,
    optimizer=ps.STLSQ(threshold),
)
print(model1)
model1.fit(x_train1, t=dt1)
x_sim1= model1.simulate(x0,t_train1)#####
mse=mean_squared_error(x_test1,x_sim1)
print('mse:',mse)
r2 = r2_score(x_test1, x_sim1)

```

```

print(f"Coeficiente de determinacin (R): {r2}")
coef1=model1.coefficients()
print(model1.coefficients())
model1.print()
print("=====")
#####
## Tcnica frente de Pareto
def Rev_Lamb(threshold_variation,x_train,dt,x_test,t_test):
    coefs=[]
    for i, threshold in enumerate(threshold_variation):
        opt=ps.STLSQ(threshold=threshold)
        model=ps.SINDy(feature_names=feature_names,optimizer=opt)
        model.fit(x_train,t=dt) #x_train
        coefs.append(model.coefficients())
    plot_pareto(coefs,opt,model,
                threshold_variation, x_test, t_test)
threshold_variation=np.linspace(0,1,21)
print(threshold_variation)
###Se generan las grficas (MSE vs lambda) para
###eleccin de lambda ptimo
Rev_Lamb(threshold_variation,x_train1,dt1,x_test1,t_train1)

```


Series de tiempo del sistema generalizado de Hénon-Heiles

Se presentan a continuación las series temporales para los casos $N = 3$, $N = 4$, para las condiciones iniciales presentadas en los Cuadros 7.1 y 7.2, respectivamente, para los diferentes valores de energía $\frac{1}{4}E_c$, $\frac{3}{4}E_c$, $\frac{99}{100}E_c$.

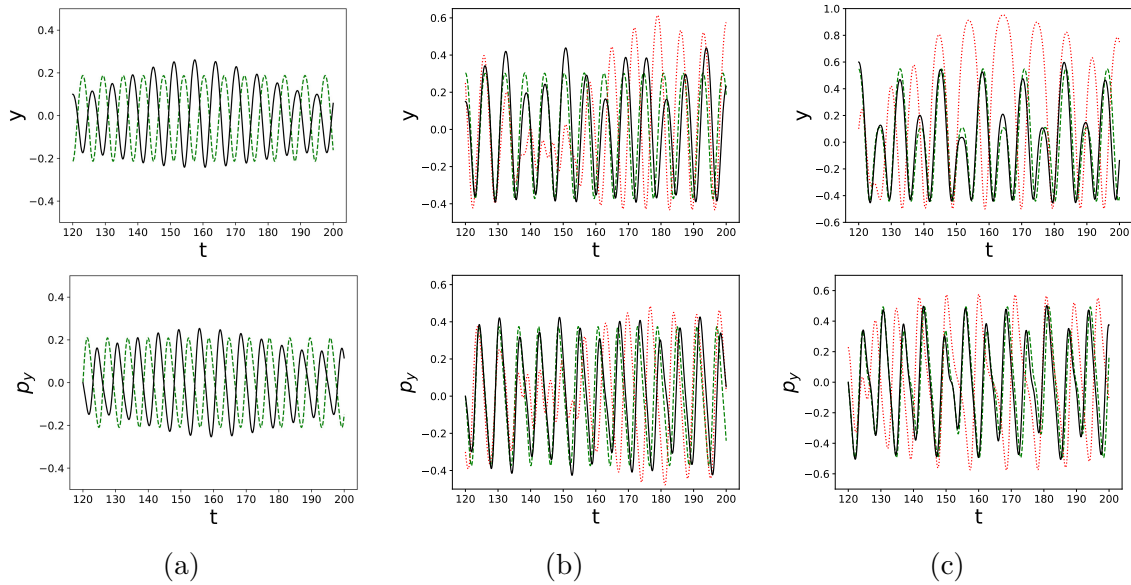


Figura F.1: Caso $N = 3$: series de tiempo para las variables dinámicas $y(t)$ y $p_y(t)$ de $\mathcal{H}_{N=3}$ (7.15) para las tres trayectorias representativas, periódica (línea discontinua verde), cuasiperiódica (línea solida negra) y caótica (línea roja punteada) de diferentes valores de energía. Aquí $E = \frac{1}{4}E_c$ en (a), $E = \frac{3}{4}E_c$ en (b), y $E = \frac{99}{100}E_c$ en (c). Estas series corresponden a las condiciones iniciales $(x_0, y_0, p_{x_0}, p_{y_0})$ usadas en la Fig. 7.7.

Planos (y, p_y) de las trayectorias seleccionadas para la reconstrucción con SINDy

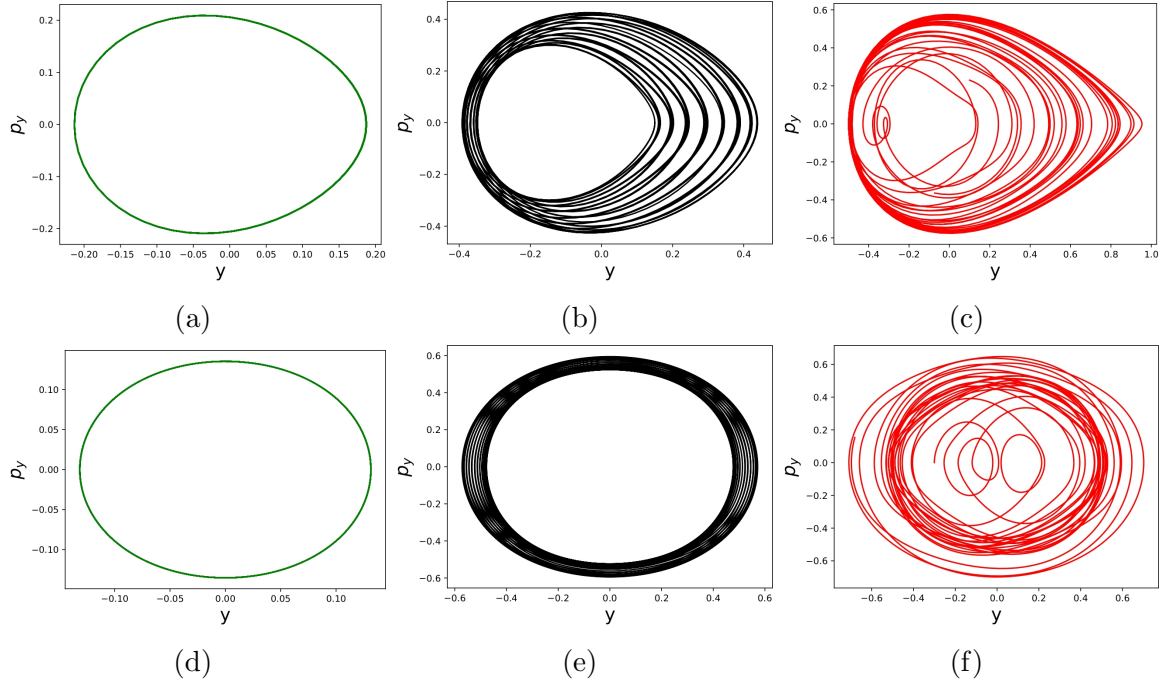
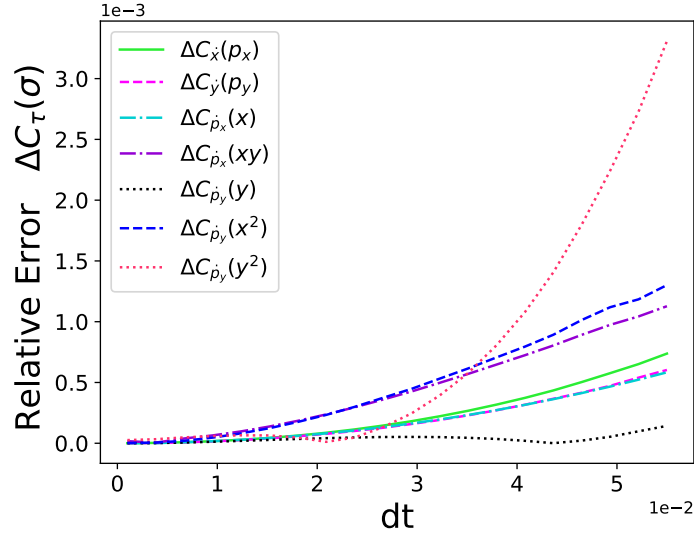
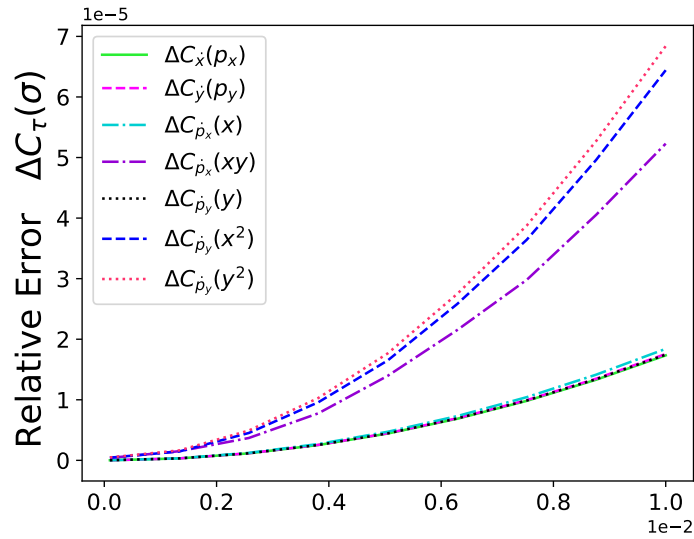


Figura F.2: Casos $N = 3$ (arriba) y $N = 4$ (abajo): trayectorias numéricas $p_y = p_y(y)$, correspondientes a las mostradas en la Fig. 7.7, para las órbitas representativas periódica (línea punteada verde), cuasiperiódica (línea negra sólida) y caótica (línea roja punteada). Aquí $E = \frac{1}{4}E_c$ en (a) y (d), $E = \frac{3}{4}E_c$ en (b) y (e), mientras que para $E = \frac{99}{100}E_c$ en (c) y (f).

Gráficas del error relativo $\Delta C_\tau(\sigma)$ en función del parámetro dt



(a)



(b)

Figura F.3: Caso $N = 3$. El error relativo $\Delta C_\tau(\sigma)$, (7.31), de los coeficientes obtenidos con SINDy en la reconstrucción de las ecuaciones de Hamilton; (a) trayectoria periódica y (b) cuasiperiódica; el paso de integración crítico es $dt_c = 0,01763$ y $dt_c = 0,02628$, respectivamente. Estas trayectorias corresponden al valor de la energía $E = \frac{1}{4}E_c$.

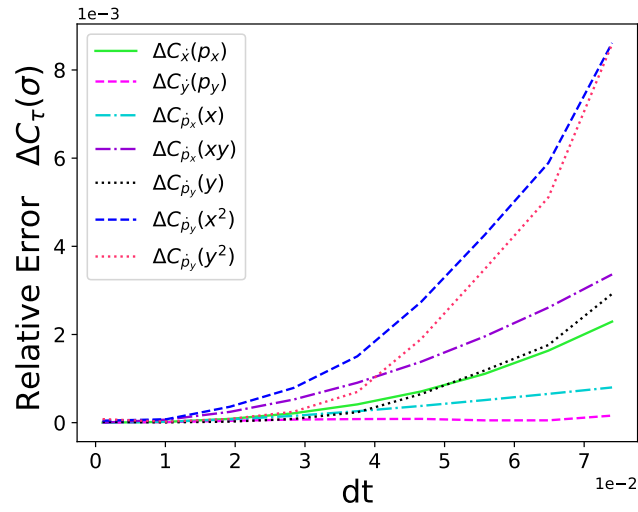
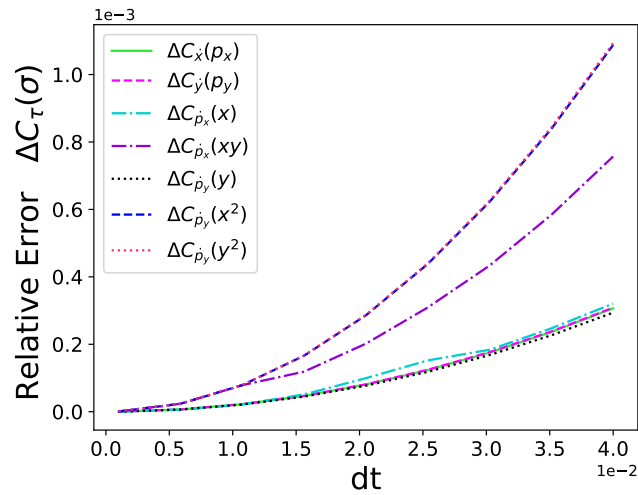
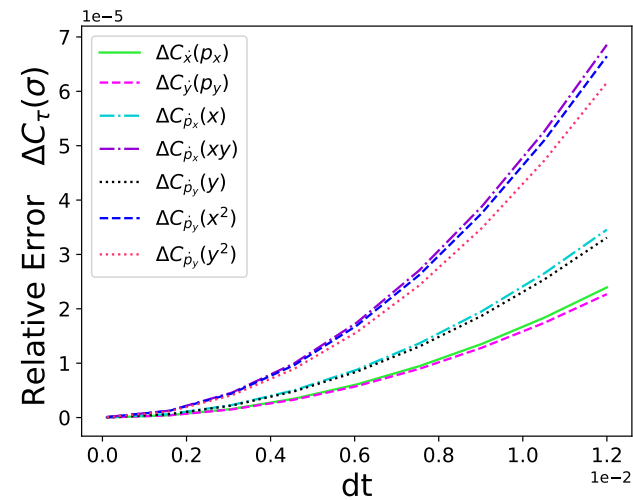
(a) $dt_c = 0,01925$ (b) $dt_c = 0,02726$ (c) $dt_c = 0,13303$

Figura F.4: Caso $N = 3$. Error relativo $\Delta C_\tau(\sigma)$ de los coeficientes obtenidos con SINDy para las ecuaciones de Hamilton de las trayectorias (a) periódica, (b) cuasiperiódica y (c) caótica, con pasos de integración críticos $dt_c = 0,01925$, $dt_c = 0,02726$ y $dt_c = 0,1303$, respectivamente. Estas trayectorias corresponden al valor de energía $E = \frac{3}{4}E_c$.

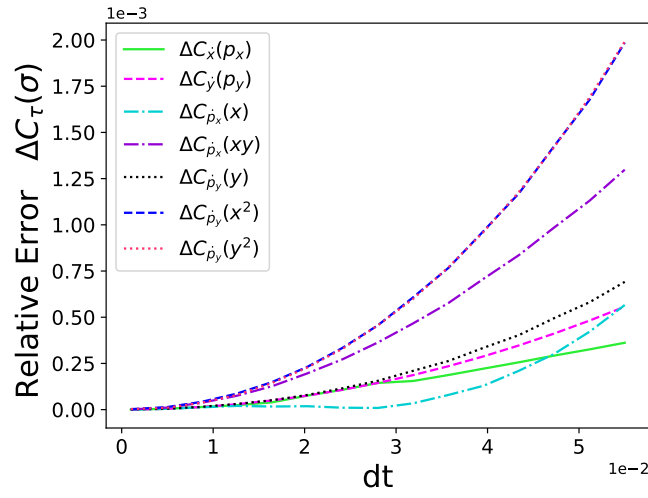
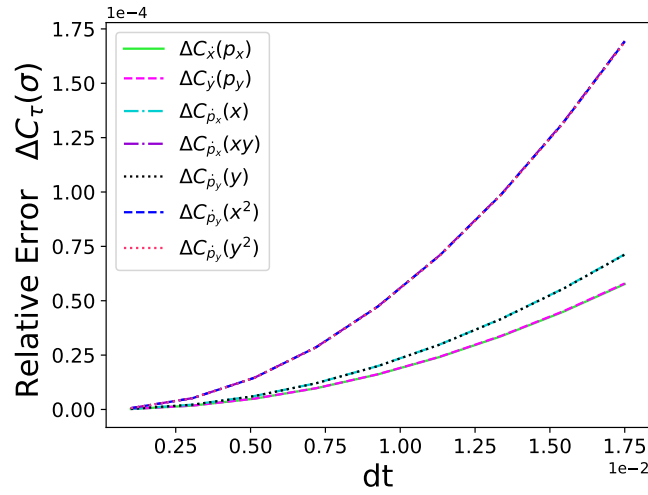
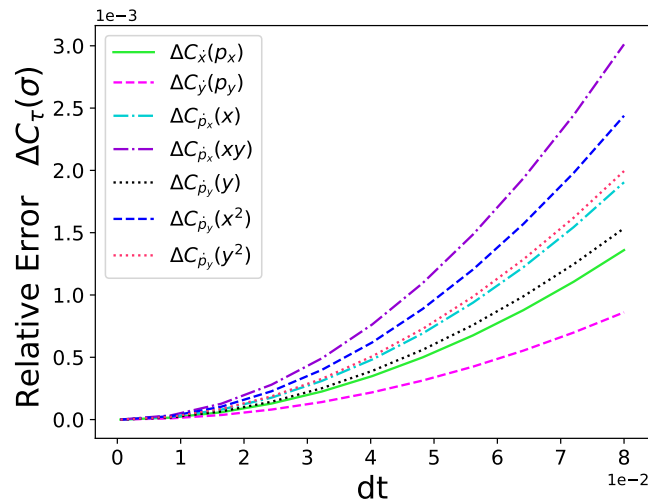
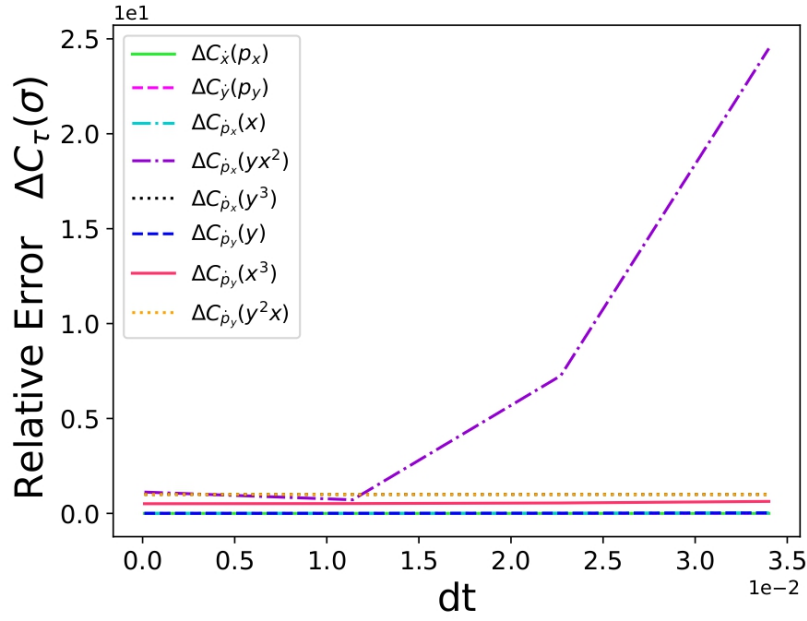
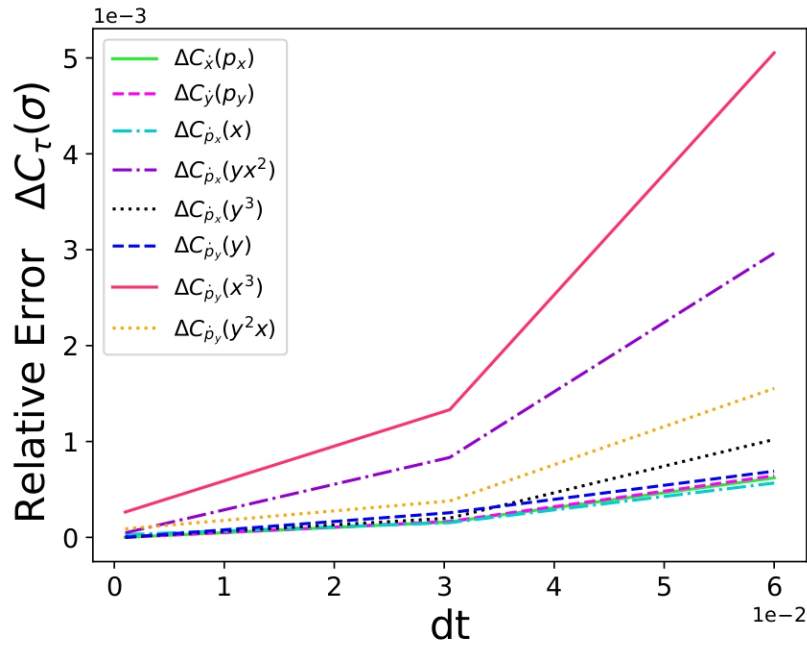
(a) $dt_c = 0,02120$ (b) $dt_c = 0,02406$ (c) $dt_c = 0,08443$

Figura F.5: Caso $N = 3$. Error relativo $\Delta C_\tau(\sigma)$ de los coeficientes obtenidos con SINDy para las ecuaciones de Hamilton de las trayectorias (a) periódica, (b) cuasiperiódica y (c) caótica con pasos de integración críticos $dt_c = 0,02120$, $dt_c = 0,02406$ y $dt_c = 0,08443$, respectivamente. Estas trayectorias corresponden al valor de energía $E = \frac{99}{100}E_c$.



(a)



(b)

Figura F.6: Caso $N = 4$. El error relativo $\Delta C_\tau(\sigma)$, de los coeficientes obtenidos por SINDy para las ecuaciones de Hamilton reconstruidas; (a) trayectoria periódica y (b) caso cuasi-periódico. Estas trayectorias corresponden al valor de energía $E = \frac{1}{4}E_c$. El incremento de tiempo crítico para la trayectoria periódica es cero, pero para el caso cuasi-periódico se da por $dt_c = 0,0305$.

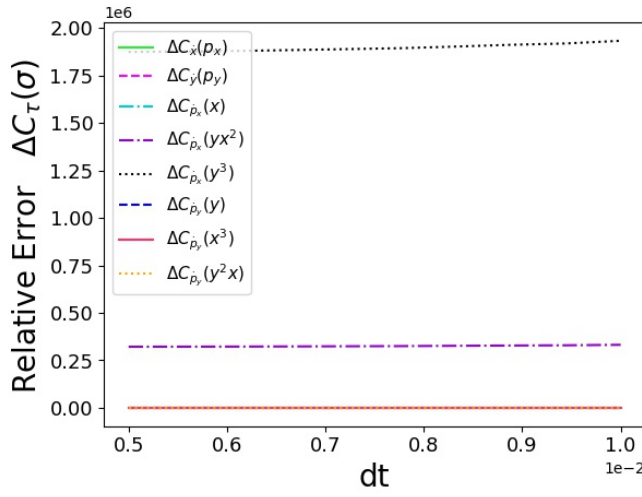
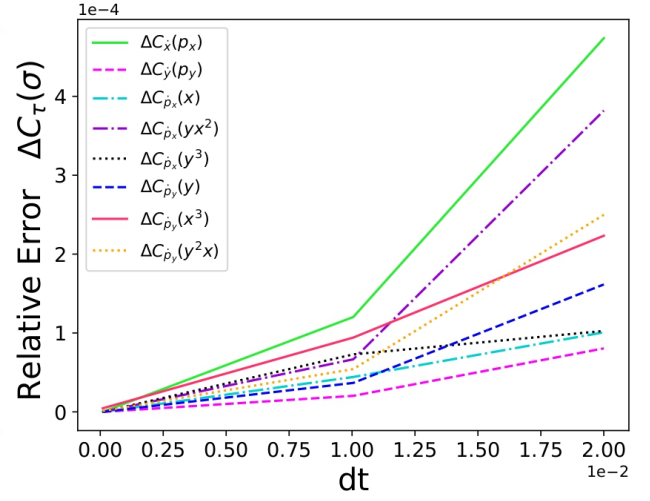
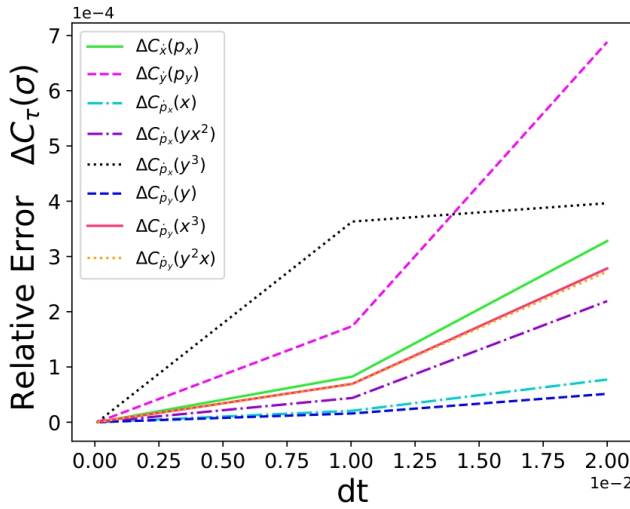
(a) $dt_c = 0$ (b) $dt_c = 0,0136$ (c) $dt_c = 0,01005$

Figura F.7: Caso $N = 4$. El error relativo $\Delta C_\tau(\sigma)$, de los coeficientes obtenidos por SINDy para las ecuaciones de Hamilton reconstruidas; (a) trayectoria periódica, (b) cuasi-periódica y (c) movimiento caótico. Para (a), el incremento de tiempo crítico es cero, mientras que para (b) y (c), son $dt_c = 0,0136$ y $dt_c = 0,01005$, respectivamente. Estas tres trayectorias corresponden al valor de energía $E = \frac{3}{4}E_c$.



Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

ACTA DE EXAMEN DE GRADO

No. 00135

Matrícula: 2222800171

Análisis y estudio de sistemas dinámicos con herramientas de machine learning

En la Ciudad de México, se presentaron a las 13:00 horas del día 31 del mes de mayo del año 2024 en la Unidad Iztapalapa de la Universidad Autónoma Metropolitana, los suscritos miembros del jurado:

DR. JOSE LUIS DEL RIO CORREA
DR. MARIO ALAN QUIROZ JUAREZ
DR. ADRIAN MAURICIO ESCOBAR RUIZ

Bajo la Presidencia del primero y con carácter de Secretario el último, se reunieron para proceder al Examen de Grado cuya denominación aparece al margen, para la obtención del grado de:

MAESTRO EN CIENCIAS (FÍSICA)

DE: FIDEL MONTOYA MOLINA

y de acuerdo con el artículo 78 fracción III del Reglamento de Estudios Superiores de la Universidad Autónoma Metropolitana, los miembros del jurado resolvieron:

Aprobar

Acto continuo, el presidente del jurado comunicó al interesado el resultado de la evaluación y, en caso aprobatorio, le fue tomada la protesta.



FIDEL MONTOYA MOLINA
ALUMNO

REVISÓ

MTRA. ROSALÍA SERRANO DE LA PAZ
DIRECTORA DE SISTEMAS ESCOLARES

DIRECTOR DE LA DIVISIÓN DE CBI

DR. ROMAN LINARES ROMERO

PRESIDENTE

DR. JOSE LUIS DEL RIO CORREA

VOCAL

DR. MARIO ALAN QUIROZ JUAREZ

SECRETARIO

DR. ADRIAN MAURICIO ESCOBAR RUIZ