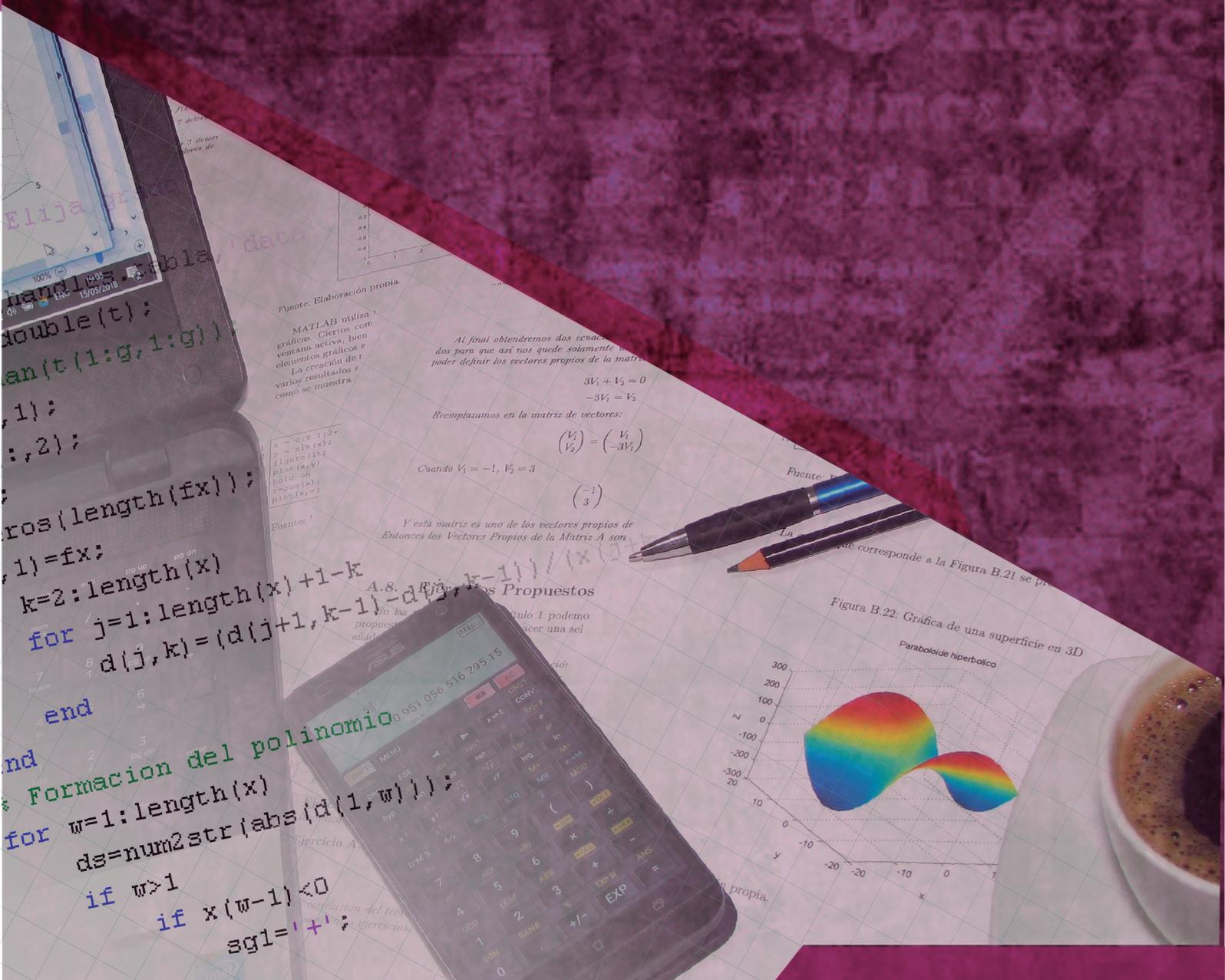


# MÉTODOS NUMÉRICOS PARA EL ANÁLISIS MATEMÁTICO CON MATLAB

JUAN F. MAÑAS-MAÑAS / MARITZA A. PINTA



Primera edición en español, 2018

Este texto ha sido sometido a un proceso de evaluación por pares externos con base en la normativa editorial de la UTMACH

---

Ediciones UTMACH

Gestión de proyectos editoriales universitarios

306 pag; 22X19cm - (Colección REDES 2017)

Título: Métodos Numéricos para el Análisis Matemático con Matlab.

Juan F. Mañas Mañas / Maritza A. Pinta (Coordinadores)

ISBN: 978-9942-24-104 -7

*Publicación digital*

---

**Título del libro:** Métodos Numéricos para el Análisis Matemático con Matlab.

ISBN: 978-9942-24-104 -7

**Comentarios y sugerencias:** [editorial@utmachala.edu.ec](mailto:editorial@utmachala.edu.ec)

**Diseño de portada:** MZ Diseño Editorial

**Diagramación:** MZ Diseño Editorial

**Diseño y comunicación digital:** Jorge Maza Córdova, Ms.

© Editorial UTMACH, 2018

© Juan Mañas / Maritza Pinta, por la coordinación

D.R. © UNIVERSIDAD TÉCNICA DE MACHALA, 2018

Km. 5 1/2 Vía Machala Pasaje

[www.utmachala.edu.ec](http://www.utmachala.edu.ec)

Machala - Ecuador

Advertencia: "Se prohíbe la reproducción, el registro o la transmisión parcial o total de esta obra por cualquier sistema de recuperación de información, sea mecánico, fotoquímico, electrónico, magnético, electro-óptico, por fotocopia o cualquier otro, existente o por existir, sin el permiso previo por escrito del titular de los derechos correspondientes".



César Quezada Abad, Ph.D  
**Rector**

Amarilis Borja Herrera, Ph.D  
**Vicerrectora Académica**

Jhonny Pérez Rodríguez, Ph.D  
**Vicerrector Administrativo**

#### **COORDINACIÓN EDITORIAL**

Tomás Fontaines-Ruiz, Ph.D  
**Director de investigación**

Karina Lozano Zambrano, Ing.  
**Jefe Editor**

Elida Rivero Rodríguez, Ph.D  
Roberto Aguirre Fernández, Ph.D  
Eduardo Tusa Jumbo, Msc.  
Irán Rodríguez Delgado, Ms.  
Sandy Soto Armijos, M.Sc.  
Raquel Tinóco Egas, Msc.  
Gissela León García, Mgs.  
Sixto Chilliquinga Villacis, Mgs.

#### **Consejo Editorial**

Jorge Maza Córdova, Ms.  
Fernanda Tusa Jumbo, Ph.D  
Karla Ibañez Bustos, Ing.  
**Comisión de apoyo editorial**

# Métodos Numéricos para el Análisis Matemático con MATLAB

Juan F. Mañas-Mañas  
Maritza A. Pinta

Coordinadores





# Resumen

El objetivo principal de esta obra es introducir al lector en el estudio de los **Métodos Numéricos** utilizando la plataforma computacional MATLAB, e impulsar en el estudiante el desarrollo de habilidades para el Análisis Matemático. Este libro es un trabajo conjunto de docentes de la Universidad Técnica de Machala (Ecuador) y de la Universidad de Almería (España) empeñados en difundir los Métodos Numéricos.

Los Métodos Numéricos son de gran importancia porque constituyen hoy en día en una herramienta fundamental para la solución de muchos problemas de las ciencias cuya solución exacta no es alcanzable y, es por tanto, necesario obtener una solución aproximada. Por otro lado, el desarrollo computacional actual, a través de software como MATLAB, permite resolver rápidamente mediante la implementación de algoritmos numéricos eficientes problemas que antes tenían una solución numérica poco factible, debido al tiempo que se requería para ello.

Este libro se compone de 6 capítulos. En el primero de ellos, se realizará un repaso de las bases matemáticas necesarias para la comprensión de los temas a tratar en los capítulos subsiguientes; se abordarán conceptos referentes a: funciones, derivación, integración, ecuaciones diferenciales, matrices y vectores.

MATLAB se fundamenta en cuatro paradigmas básicos de la programación: la programación secuencial, la programación estructurada, la programación modular y la programación orientada a objetos. Por ello se ha considerado necesario en el segundo capítulo hacer una presentación de la sintaxis de comandos secuenciales que sean de relevancia significativa para la construcción de algoritmos de métodos numéricos y la visualización de resultados. Para ello, se empieza con la caracterización de las variables en el entorno de MATLAB; luego, se presentan los comandos que implementan diferentes funciones matemáticas. De esta forma, se aprovecha los recursos disponibles en este paquete computacional para fortalecer las soluciones numéricas de nuestros algoritmos.

El capítulo 3 nos introduce en el estudio de la derivación numérica de funciones, la cual tiene muchas aplicaciones, especialmente en la resolución numérica de ecuaciones diferenciales. Además, nos permite determinar la derivada de un orden determinado de una función en un punto dado, utilizando solamente los valores que toma la función en una serie de puntos. Se realizará un estudio con detalle pero con enfoque práctico de estas fórmulas y del error teórico cometido, prestando atención a un método relevante de aceleración de la convergencia conocido como método de extrapolación de Richardson; haciendo la implementación de estos métodos en MATLAB a través de ejemplos prácticos.

Posteriormente los capítulos 4 y 5 están dedicados a la integración numérica

que tienen como objetivo aproximar numéricamente integrales definidas, las cuales tienen muchas aplicaciones tanto en matemáticas como en procesos científicos-técnicos. Este cálculo suele ser complicado y en la mayoría de los casos es inviable si se pretende expresar el valor de la integral definida como la evaluación de combinación de funciones elementales.

El capítulo 4 tiene por objetivo obtener expresiones, usualmente denominadas fórmulas de cuadratura, que permitan aproximar de la forma más exacta posible una integral definida. Obtendremos fórmulas basadas en polinomios interpoladores, denominadas fórmulas de Newton-Cotes. Como es habitual en el análisis numérico proporcionaremos expresiones para el error cometido al usar estas fórmulas. Adicionalmente aplicaremos el proceso de aceleración o método de Romberg y algunas breves notas sobre cuadraturas adaptativas.

En el capítulo 5, dedicado a la integración numérica, se presentarán las fórmulas de cuadratura gaussianas. La ventaja de estas fórmulas es que los nodos involucrados no son fijos, sino que van a ser los ceros de determinados polinomios ortogonales; de esta forma natural, imbricamos la integración numérica con la Teoría de Aproximación a través del uso de los polinomios ortogonales. Haremos un estudio detallado y práctico de las fórmulas gaussianas, analizando su exactitud máxima, el error y el cálculo eficiente de sus nodos y pesos mediante los valores y vectores propios de la matriz de Jacobi.

Nuestro último capítulo está dedicado a las ecuaciones diferenciales, que es la herramienta matemática más útil a la hora de describir problemas en todos los ámbitos de las ciencias y también en otras ramas del conocimiento. Es bien conocido su uso en la modelización matemática en biología, ingeniería, medicina, informática y cualquier área científico-técnica, pero también en otras áreas como en el estudio de comportamientos sociales o en economía.

En este libro introductorio a los métodos numéricos pretendemos acercar al lector a la resolución numérica de ecuaciones diferenciales ordinarias (EDO). Las ecuaciones, o de forma más natural, los sistemas de EDO que aparecen en la modelación matemática, raramente son resolubles utilizando solamente el análisis matemático y es imprescindible el uso del análisis numérico. Se asumirá que el lector posee un cierto conocimiento de EDO, y se presentarán de forma práctica métodos útiles de resolución numérica de problemas de valores iniciales y problemas de contorno. Se prestará atención a los denominados problemas stiff. La aplicación de los métodos presentados en este capítulo hará uso necesariamente del ordenador y del programa MATLAB.

# Listado de autores por capítulos

- Capítulo 1:
  - Maritza A. Pinta.
  - Fausto F. Redrován.
- Capítulo 2:
  - Eduardo Tusa.
  - Carlos Loor.
- Capítulo 3:
  - Juan F. Mañas-Mañas.
  - Maritza A. Pinta.
- Capítulo 4:
  - Juan F. Mañas-Mañas.
  - Maritza A. Pinta.
- Capítulo 5:
  - Juan F. Mañas-Mañas.
  - Maritza A. Pinta.
- Capítulo 6:
  - Juan J. Moreno-Balcázar.



# Dedicatorias

*Dedico esta obra a mi amado esposo Egipto y a mi hijo Juan José, por su amor y apoyo incondicional.*

**Maritza A. Pinta**

*Esta obra es dedicada a mis estudiantes, quienes han tenido la libertad de compartirme sus inquietudes, inconformidades y preocupaciones; en torno al área de la programación y métodos numéricos en MATLAB. Dedico este trabajo a toda mi familia, mis padres, mis hermanas, mis sobrinos; quienes no han dudado en demostrarme su enorme afecto, cariño y paciencia en cada acierto y desacierto que se concibe a lo largo de mi vida. A aquellos grandes amigos y amigas que a pesar del tiempo, no han renunciado a caminar junto a mi lado. Finalmente, a todos los grandes soñadores del mundo que se levantan cada mañana dispuestos a cambiar el curso de su destino.*

**Eduardo Tusa**

*A toda mi familia. En especial, a mi mujer y compañera, Loli, y a mis hijas, Rocío y Mar, por ser tan maravillosas y a mis padres, Juan José y Rosa, gracias a ellos empezó todo, siempre serán mis referentes.*

**Juan J. Moreno-Balcázar**

*Dedico esta obra a mis padres, Francisco y Rosalía, ya que gracias a ellos he llegado hasta donde estoy hoy en día. A mi hermana Rosa María y mi novia Mónica por el apoyo recibido, y a toda mi familia y amigos por los ánimos dados.*

**Juan F. Mañas-Mañas**

*A mi esposa Rosita, mi consejera e inspiración. A mis hijos Abigaíl y Mateo, mi motivación y razón. A mis padres Fausto y Mirian, mi guía y apoyo. A la memoria de mi abuelito Fausto, mi primer profesor de matemáticas.*

**Fausto F. Redrován**



# Agradecimientos

Los autores compiladores, Maritza A. Pinta y Juan F. Mañas-Mañas agradecen en primer lugar a todos los autores su trabajo, perseverancia y todo el esfuerzo dedicado a la hora de realizar este libro, ya que gracias a ellos el mismo ha sido posible.

Todos los autores aprovechamos estas líneas para agradecer a las principales autoridades de la Universidad Técnica de Machala, quienes han brindado el tiempo, los recursos y el escenario para impulsar la producción científica de sus principales actores. Asimismo, expresamos nuestra gratitud a la Editorial UTMACH y al Centro de Investigaciones por fortalecer iniciativas orientadas a preservar el legado científico de los investigadores.

Los autores de la Universidad Técnica de Machala expresamos también gratitud al decano, subdecano, coordinadores y colegas docentes de la Unidad Académica de Ingeniería de Civil, por su cooperación en potenciar los espacios de investigación alcanzados y a los miembros del Grupo de Investigación de Automatización, Matemática y Tecnologías de la Información y las Comunicaciones, AutoMath-TIC.

Por parte de la Universidad de Almería, el autor Juan J. Moreno Balcázar está parcialmente financiado por los siguientes proyectos e instituciones: Grupo de Investigación FQM-229 de la Junta de Andalucía adscrito al Campus de Excelencia Internacional del Mar (CEIMAR); Ministerio de Economía y Competitividad de España y el Fondo Europeo de Desarrollo Regional (FEDER) con el proyecto MTM2014-53963-P; Centro para el Desarrollo y la Transferencia de la Innovación Matemática en la Empresa (CDTIME). El autor Juan F. Mañas-Mañas está financiado por una beca predoctoral del Plan Propio de la Universidad de Almería, además, parcialmente financiado por los siguientes proyectos e instituciones: Grupo de Investigación FQM-229 de la Junta de Andalucía adscrito al Campus de Excelencia Internacional del Mar (CEIMAR); Ministerio de Economía y Competitividad de España y el Fondo Europeo de Desarrollo Regional (FEDER) con el proyecto MTM2014-53963-P.

Finalmente, y no por ello menos importante, agradecemos también a los árbitros anónimos la revisión de los capítulos, gracias a sus comentarios y aportaciones esta obra ha ganado valor y relevancia.



# Índice general

<b>1. CONCEPTOS PRELIMINARES</b>	<b>1</b>
1.1. Teoremas importantes para funciones continuas. . . . .	1
1.1.1. Conservación del signo de las funciones continuas . . . . .	2
1.1.2. Teorema de Bolzano . . . . .	3
1.1.3. Teorema del valor intermedio (o Propiedad de Darboux) . . . . .	5
1.2. Derivación . . . . .	6
1.2.1. Interpretación Geométrica de la Derivada . . . . .	6
1.2.2. Fórmulas de derivación . . . . .	7
1.2.3. Segunda Derivada y Derivadas Sucesivas . . . . .	8
1.3. Valor extremo . . . . .	9
1.4. Teorema de Rolle. . . . .	10
1.5. Teorema del Valor Medio. . . . .	12
1.6. Integración. . . . .	13
1.6.1. Integral Definida . . . . .	14
1.6.2. Teoremas Fundamentales del Cálculo . . . . .	14
1.6.3. Integración Por Partes . . . . .	15
1.7. Ecuaciones Diferenciales. . . . .	15
1.7.1. Tipos de Ecuaciones Diferenciales . . . . .	15
1.7.2. Solución de una Ecuación Diferencial . . . . .	16
1.7.3. Ecuaciones Diferenciales de Primer Orden . . . . .	16
1.7.4. Ecuaciones Diferenciales Lineales Homogéneas con Coefi- cientes Constantes de $n$ -ésimo Orden. . . . .	17
1.8. Matrices. . . . .	18
1.8.1. Tipos de Matrices . . . . .	18
1.8.2. Operaciones entre Matrices . . . . .	18
1.8.3. Determinantes . . . . .	20
1.8.4. Inversa de una matriz . . . . .	22
1.8.5. Valores Propios (AutoValores) y Vectores Propios (Auto- Vectores) . . . . .	24
<b>2. INTRODUCCIÓN A MATLAB</b>	<b>27</b>
2.1. Introducción . . . . .	27
2.2. Entorno de MATLAB . . . . .	28
2.3. Variables . . . . .	30
2.3.1. Tipo de variables . . . . .	30
2.3.2. Jerarquía de operaciones aritméticas . . . . .	31
2.4. Vectores . . . . .	32

2.4.1.	Ingreso de vectores . . . . .	32
2.4.2.	Operaciones con vectores . . . . .	34
2.5.	Matrices . . . . .	36
2.5.1.	Ingreso de matrices . . . . .	36
2.5.2.	Operaciones con matrices . . . . .	38
2.6.	Polinomios . . . . .	39
2.6.1.	Ingreso de polinomios . . . . .	40
2.6.2.	Operaciones con polinomios . . . . .	40
2.7.	Script de MATLAB . . . . .	42
2.8.	Estructuras de control . . . . .	43
2.8.1.	Estructuras de decisión . . . . .	44
2.8.2.	Estructuras de repetición . . . . .	46
2.9.	Funciones . . . . .	48
2.9.1.	Funciones en línea . . . . .	48
2.9.2.	Funciones modulares . . . . .	50
<b>3.</b>	<b>DERIVACIÓN NUMÉRICA</b>	<b>53</b>
3.1.	Serie de Taylor . . . . .	53
3.2.	Error de truncamiento en la serie de Taylor . . . . .	56
3.3.	Fórmulas de derivación numérica basadas en la Serie de Taylor . . . . .	56
3.3.1.	Fórmulas progresivas o de diferencias divididas finitas hacia delante . . . . .	57
3.3.2.	Formulas regresivas o de diferencias divididas finitas hacia atrás . . . . .	58
3.3.3.	Fórmulas de diferencias finitas centradas . . . . .	60
3.4.	Derivadas de orden superior . . . . .	62
3.5.	Fórmulas basadas en polinomios de interpolación . . . . .	68
3.5.1.	Utilizando el polinomio interpolador de Lagrange . . . . .	69
3.5.2.	El polinomio interpolador de Newton . . . . .	71
3.6.	Inestabilidad de las fórmulas de derivación numérica . . . . .	72
3.7.	Extrapolación de Richardson . . . . .	73
<b>4.</b>	<b>CUADRATURAS DE NEWTON-COTES</b>	<b>79</b>
4.1.	Introducción a la integración numérica . . . . .	80
4.2.	Método de coeficientes indeterminados . . . . .	81
4.2.1.	Regla del Rectángulo a Izquierda . . . . .	81
4.2.2.	Regla del Rectángulo a Derecha . . . . .	82
4.2.3.	Regla del Punto Medio . . . . .	83
4.2.4.	Regla de los Trapecios . . . . .	84
4.2.5.	Regla de Simpson . . . . .	85
4.2.6.	Método de coeficientes indeterminados . . . . .	86
4.3.	Fórmulas de cuadratura interpolatorias . . . . .	87
4.3.1.	Análisis del error . . . . .	88
4.4.	Fórmulas de Newton-Cotes . . . . .	92
4.4.1.	Fórmulas de Newton-Cotes simples . . . . .	92
4.4.2.	Fórmulas de Newton-Cotes Compuestas . . . . .	94
4.5.	Método de Romberg . . . . .	100

4.6.	Cuadraturas adaptativas . . . . .	101
4.6.1.	Trapezios Adaptativo . . . . .	102
4.6.2.	Simpson Adaptativo . . . . .	102
<b>5.</b>	<b>CUADRATURAS GAUSSIANAS</b>	<b>107</b>
5.1.	Motivación . . . . .	108
5.2.	Introducción a polinomios ortogonales . . . . .	109
5.3.	Familias clásicas de polinomios ortogonales . . . . .	111
5.3.1.	Polinomios ortogonales de Jacobi . . . . .	111
5.3.2.	Polinomios ortogonales de Laguerre . . . . .	112
5.3.3.	Polinomios ortogonales de Hermite . . . . .	112
5.4.	Fórmulas de cuadratura . . . . .	113
5.5.	Algoritmo para cuadraturas gaussianas . . . . .	115
5.5.1.	Paso 1: Transformar la integral convenientemente . . . . .	116
5.5.2.	Paso 2: Calculo de los nodos $\mathbf{x}_k$ . . . . .	117
5.5.3.	Paso 3: Calculo de los pesos $\lambda_k$ . . . . .	119
5.5.4.	Ejemplo . . . . .	119
5.6.	Error de las cuadraturas gaussianas . . . . .	121
5.7.	Implementación de las cuadraturas gaussianas . . . . .	123
5.7.1.	Código para cuadraturas gaussianas . . . . .	125
5.7.2.	Utilización del código . . . . .	126
<b>6.</b>	<b>ECUACIONES DIFERENCIALES ORDINARIAS</b>	<b>131</b>
6.1.	Definiciones básicas . . . . .	131
6.1.1.	Problemas de valores iniciales . . . . .	132
6.1.2.	Existencia y unicidad de solución de PVI . . . . .	133
6.1.3.	Estabilidad de un PVI . . . . .	134
6.2.	Número de condición de un PVI . . . . .	136
6.3.	Métodos de un paso para la resolución numérica de PVI . . . . .	138
6.3.1.	Un primer acercamiento: el método de Euler . . . . .	138
6.3.2.	Métodos de un paso: consistencia, estabilidad, convergencia y orden . . . . .	140
6.3.3.	Problemas stiff. A–estabilidad . . . . .	142
6.3.4.	Métodos de Runge–Kutta . . . . .	144
6.3.5.	Algunos métodos RK . . . . .	146
6.4.	Métodos multipaso lineales para la resolución de PVI . . . . .	148
6.4.1.	Generación de los métodos multipaso: métodos de Adams y métodos de diferencias regresivas . . . . .	151
6.4.2.	Orden y A–estabilidad de los métodos multipaso lineales . . . . .	152
<b>A.</b>	<b>Apéndice A</b>	<b>157</b>
A.1.	Derivación . . . . .	157
A.2.	Valor Extremo . . . . .	159
A.3.	Teorema Rolle . . . . .	160
A.4.	Teorema Valor Medio . . . . .	161
A.5.	Integración . . . . .	162
A.5.1.	Integración Por Partes . . . . .	164

A.5.2. Teoremas Fundamentales del Cálculo . . . . .	165
A.6. Ecuaciones Diferenciales . . . . .	165
A.6.1. Solución de una Ecuación Diferencial . . . . .	166
A.6.2. Ecuaciones Diferenciales de Primer Orden . . . . .	166
A.6.3. Ecuaciones Diferenciales Lineales Homogéneas con Coefi- cientes Constantes de Enésimo Orden. . . . .	166
A.7. Matrices . . . . .	166
A.7.1. Tipo de Matrices . . . . .	167
A.7.2. Adición y sustracción de matrices. . . . .	170
A.7.3. Multiplicación de Matrices . . . . .	170
A.7.4. Determinantes . . . . .	171
A.7.5. Inversa de una matriz . . . . .	171
A.7.6. Valores Propios (Autovalores) y Vectores Propios (Auto- Vectores) . . . . .	173
A.8. Ejercicios Propuestos . . . . .	176
<b>B. Apéndice B</b>	<b>179</b>
B.1. Cuestiones resueltas . . . . .	179
B.1.1. $4 * 2 + (3 * 3 + 2) \wedge 2 - 9 \wedge 2$ . . . . .	179
B.1.2. $7 \wedge 2 + 3 * 4 + (2 * 3 + 5) \wedge 2$ . . . . .	179
B.1.3. Seleccione un nombre de variable que no sea reconocido en MATLAB: . . . . .	180
B.1.4. Seleccione un nombre de variable que no sea reconocido en MATLAB: . . . . .	180
B.1.5. ¿Cuál de las siguientes variables es tipo character? . . . . .	180
B.1.6. ¿Cuál de las siguientes variables es tipo lógico?: . . . . .	180
B.2. Matrices . . . . .	181
B.3. Representaciones Gráficas . . . . .	183
B.3.1. Comandos para Gráficos 2D . . . . .	183
B.3.2. Comandos para Gráficos 3D . . . . .	189
B.4. Programación con MATLAB . . . . .	192
B.4.1. Programación Secuencial . . . . .	192
B.4.2. Estructuras Condicionales . . . . .	194
B.4.3. Estructura Repetitivas . . . . .	198
B.5. Funciones . . . . .	201
<b>C. Apéndice C</b>	<b>205</b>
C.1. Ejemplo de la Serie de Taylor. . . . .	205
C.2. Grado de exactitud . . . . .	206
C.3. Derivación numérica utilizando fórmulas de Taylor. . . . .	207
C.4. Fórmula de Richardson . . . . .	224
C.5. Ejercicios propuestos . . . . .	226
<b>D. Apéndice D</b>	<b>229</b>
D.1. Implementaciones secciones 4.2.1-4.2.5. . . . .	229
D.2. Fórmulas de cuadratura compuestas . . . . .	230
D.2.1. Rectángulo a Izquierda Compuesto . . . . .	231

D.2.2. Rectángulo a Derecha Compuesto . . . . .	231
D.2.3. Punto Medio Compuesto . . . . .	231
D.2.4. Trapecios Compuesto . . . . .	231
D.2.5. Simpson Compuesto . . . . .	231
D.3. Ejemplos resueltos . . . . .	231
D.4. Método de Romberg . . . . .	244
D.5. Ejemplo de Cuadratura de Trapecios Adaptativa . . . . .	245
D.6. Ejercicios propuestos . . . . .	246
<b>E. Apéndice E</b>	<b>251</b>
E.1. Ejemplos resueltos . . . . .	251
E.2. Ejercicios propuestos . . . . .	257
<b>F. Apéndice F</b>	<b>263</b>
F.1. Ejemplos (ejercicios resueltos) . . . . .	263
F.2. Implementación en Matlab de los métodos de Runge–Kutta y de Adams–Basforth. Órdenes <b>ode</b> . . . . .	280
F.2.1. Método de Runge–Kutta clásico . . . . .	281
F.2.2. Método de Adams–Bashforth de 4 pasos . . . . .	282
F.2.3. Órdenes <b>ode</b> de Matlab . . . . .	283
F.3. Ejemplos usando Matlab . . . . .	284
F.4. Ejercicios propuestos . . . . .	298
F.5. Algunas demostraciones . . . . .	302
F.5.1. Demostración del Teorema de Lax 6.2. . . . .	302
F.5.2. Demostración del teorema sobre la convergencia de los méto- dos de Runge–Kutta explícitos (Teorema 6.4). . . . .	302
F.6. Generación de los métodos de Adams . . . . .	305



# Índice de tablas

1.1. Nomenclaturas de la derivada. . . . .	7
1.2. Fórmulas de derivación. . . . .	8
1.3. Derivadas Sucesivas . . . . .	8
1.4. Clasificación de las ecuaciones diferenciales. . . . .	16
1.5. Ecuaciones diferenciales de primer orden. . . . .	17
2.1. Scripts vs Funciones . . . . .	50
3.1. Principales fórmulas de derivación numérica . . . . .	64
4.1. Tabla de principales cuadraturas simples de Newton-Cotes. . . . .	93
5.1. Tabla de Parámetros para polinomios ortogonales mónicos. . . . .	118
A.1. Fórmulas de integración inmediatas. . . . .	162



# Índice de figuras

1.1. Función Continua. . . . .	2
1.2. Función Discontinua. . . . .	2
1.3. Teorema de Conservación de Signo. . . . .	3
1.4. Teorema de Bolzano. . . . .	4
1.5. Teorema del Valor Intermedio (caso 1) . . . . .	5
1.6. Teorema del Valor Intermedio (caso 2) . . . . .	5
1.7. Interpretación Geométrica de la Derivada . . . . .	7
1.8. Valores extremos en un intervalo $[a, b]$ . . . . .	9
1.9. Máximos y mínimos en un intervalo abierto $(a, b)$ . . . . .	10
1.10. Teorema de Rolle . . . . .	11
1.11. Función no derivable en un punto. . . . .	11
1.12. Teorema del Valor Medio . . . . .	13
2.1. Entorno de la interface de MATLAB o Ventana Principal . . . . .	28
2.2. Barra de herramientas del Current Folder . . . . .	28
2.3. Interface de la ventana Command Window en MATLAB . . . . .	29
2.4. Ventana del workspace con 5 tipos de variables inicializadas . . . . .	29
2.5. Tipo de variables disponibles en MATLAB . . . . .	31
2.6. Ingreso de vectores fila y columna en MATLAB . . . . .	32
2.7. Creación de un vector fila igualmente espaciado . . . . .	33
2.8. Vectores fila de unos y ceros . . . . .	33
2.9. Transpuesta de un vector fila a un vector columna . . . . .	34
2.10. Consulta de los elementos de un vector . . . . .	34
2.11. Suma de un vector con otro vector y un escalar . . . . .	35
2.12. Resta de un vector con otro vector y un escalar . . . . .	35
2.13. Producto de vectores . . . . .	35
2.14. División de vectores . . . . .	36
2.15. Potencia de vectores . . . . .	36
2.16. ingreso de una matriz $3 \times 3$ . . . . .	37
2.17. Creación de matrices $3 \times 3$ de unos y ceros . . . . .	37
2.18. Creación de una matriz identidad $3 \times 3$ . . . . .	37
2.19. Suma de matrices . . . . .	38
2.20. Resta de matrices . . . . .	38
2.21. Producto de matrices . . . . .	38
2.22. Transpuesta de una matriz . . . . .	39
2.23. Determinante de una matriz . . . . .	39
2.24. Inversa de una matriz . . . . .	39
2.25. Polinomios . . . . .	40

2.26.	Raíces de un polinomio . . . . .	40
2.27.	Polinomio a partir de las raíces . . . . .	41
2.28.	Evaluación de un polinomio . . . . .	41
2.29.	Producto de polinomios . . . . .	42
2.30.	División de polinomios . . . . .	42
2.31.	Ventana del Editor en MATLAB . . . . .	43
2.32.	Ejemplo de un script editado en MATLAB . . . . .	43
2.33.	Estructura Condicional Simple: Síntaxis en MATLAB (izquierda) y Diagrama de flujo (derecha) . . . . .	44
2.34.	Estructura Condicional Doble: Síntaxis en MATLAB (izquierda) y Diagrama de flujo (derecha) . . . . .	44
2.35.	Estructura Condicional Múltiple: Síntaxis en MATLAB (izquierda) y Diagrama de flujo (derecha) . . . . .	45
2.36.	Estructura Condicional Múltiple SWITCH/CASE: Síntaxis en MATLAB (izquierda) y Diagrama de flujo (derecha) . . . . .	45
2.37.	Estructura de Repetición Indexada : Síntaxis en MATLAB (izquierda) y Diagrama de flujo (derecha) . . . . .	46
2.38.	Estructura de Repetición Condicionada : Síntaxis en MATLAB (izquierda) y Diagrama de flujo (derecha) . . . . .	47
2.39.	Interrupción con BREAK: Síntaxis en MATLAB (izquierda) y Diagrama de flujo (derecha) . . . . .	47
2.40.	Interrupción con CONTINUE: Síntaxis en MATLAB (izquierda) y Diagrama de flujo (derecha) . . . . .	48
2.41.	Creación de la función en línea $f(x) = e^{-x^2}$ . . . . .	48
2.42.	Evaluación de la función en línea $f(x) = e^{-x^2}$ . . . . .	49
2.43.	Gráfica de la función en línea $f(x) = e^{-x^2}$ . . . . .	49
2.44.	Declaración de una función en MATLAB . . . . .	50
3.1.	Funciones $f(x)$ y $T_{4,0}[e^x - \text{sen}(x)](x)$ . . . . .	55
4.1.	Interpretación geométrica del Rectángulo a Izquierda . . . . .	82
4.2.	Interpretación geométrica del Rectángulo a Derecha . . . . .	83
4.3.	Interpretación geométrica de la Regla del Punto Medio . . . . .	83
4.4.	Interpretación geométrica de la Regla de los Trapecios . . . . .	85
6.1.	Interpretación geométrica del método de Euler . . . . .	139
A.1.	Valores máximos y mínimos de una función continua en un intervalo cerrado $[0, 3]$ . . . . .	159
A.2.	Extremos relativos en una función . . . . .	159
A.3.	Ejemplo de Aplicación del Teorema de Rolle . . . . .	160
A.4.	Ejemplo de aplicación del Teorema del Valor medio. . . . .	161
B.1.	Matriz de $3 \times 3$ de números aleatorios enteros entre 0 y 10 . . . . .	181
B.2.	Rango y norma de una matriz . . . . .	181
B.3.	Valores y vectores propios . . . . .	182
B.4.	Descomposición QR . . . . .	182
B.5.	Descomposición LU . . . . .	183

B.6. Comando plot . . . . .	183
B.7. Función $y = \text{sen}(x)$ . . . . .	184
B.8. Comando hold on . . . . .	184
B.9. Funciones $y = \text{sen}(x)$ y $z = \text{cos}(x)$ . . . . .	185
B.10. Comando hold off . . . . .	185
B.11. Gráfica de la función $w = \text{tan}(x)$ . . . . .	186
B.12. Comando plot con tres argumentos . . . . .	186
B.13. Trazada con asteriscos rojos . . . . .	187
B.14. Comandos grid, xlabel, ylabel y title . . . . .	187
B.15. Función $y = \text{sen}(x)$ con cuadrícula, título y etiquetas en los ejes .	188
B.16. Comando <b>area</b> para presentar el área bajo la curva . . . . .	188
B.17. Área bajo la curva de una función . . . . .	189
B.18. Uso del comando plot3 . . . . .	189
B.19. Gráfica de una Hélice . . . . .	190
B.20. Rejilla en el plano $xy$ para el dominio $0 \leq x \leq 5$ y $0 \leq y \leq 5$ . . .	190
B.21. Uso de los comandos meshgrid y mesh . . . . .	191
B.22. Gráfica de una superficie en 3D . . . . .	191
B.23. Rectángulo inscrito en una parábola . . . . .	192
B.24. Cálculo del área máxima de un rectángulo inscrito en una parábola	192
B.25. Ejecución del Programa de la Figura B.24 . . . . .	193
B.26. Forma de la ventana . . . . .	193
B.27. Cálculo de las dimensiones de la ventana de la Figura B.26 . . . .	194
B.28. Ejecución del Programa B.27 . . . . .	194
B.29. Descomposición de un número de 3 cifras . . . . .	195
B.30. Ejecución del Programa de la Figura B.29 . . . . .	195
B.31. Clasificación de triángulos basado en las dimensiones de sus lados	196
B.32. Ejecución del Programa de la Figura B.31 . . . . .	196
B.33. Conversion entre temperaturas Celsius y Fahrenheit . . . . .	197
B.34. Ejecución del Programa de la Figura B.33 . . . . .	197
B.35. Descomposición de factores primos . . . . .	198
B.36. Ejecución del Programa de la Figura B.35 . . . . .	198
B.37. Conteo de lanzamientos de un dado hasta obtener un 5 . . . . .	199
B.38. Ejecución del Programa de la Figura B.37 . . . . .	199
B.39. Sumatoria den números pares hasta un número divisible para 5 .	200
B.40. Ejecución del Programa de la Figura B.39 . . . . .	200
B.41. Función que calcula las raíces de un polinomio de la forma $ax^2 +$ $bx + c$ . . . . .	201
B.42. Aplicación del comando <b>help</b> para la función <b>raíces</b> y su invoca- ción desde la Ventana de Comandos. . . . .	201
B.43. Función que calcula calcula el área y el volumen de un cilindro . .	202
B.44. Aplicación del comando <b>help</b> para la función <b>calcula_cilindro</b> y su invocación desde la Ventana de Comandos. . . . .	202
B.45. Cilindro de radio $r = 5$ y altura $h = 10$ . . . . .	203
C.1. Funciones $f(x)$ y $T_{3,\pi/2}[4x \cdot \text{sen}(x)](x)$ . . . . .	206
D.1. Representación de $f''(t) = -2e^{-t}(5 \cos(5t) + 12\text{sen}(5t))$ . . . . .	234

D.2.	Representación de $ f''(t) $ .	235
D.3.	Representación de $f^{(4)}(t) = e^{-t}(480 \cos(5t) + 476 \operatorname{sen}(5t))$ .	236
D.4.	Representación de $ f^{(4)}(t) $ .	236
D.5.	Cuadratura RIC con 6 nodos	238
D.6.	Cuadratura RIC con 11 nodos	238
D.7.	Cuadratura RDC con 11 nodos	239
D.8.	Punto Medio Compuesto con 11 nodos	240
D.9.	Trapecios Compuesto con 5 nodos	241
D.10.	Trapecios Compuesto con 11 nodos	241
E.1.	Gráfica de $ f^{(10)}(t) $ en el intervalo $[-1, 1]$ .	252
E.2.	Gráfica de $ f^{(10)}(t) $ en el intervalo $[0, 10]$ .	257
F.1.	Región de $A$ -estabilidad del método de Euler modificado	266
F.2.	Región de $A$ -estabilidad del método de Runge–Kutta clásico	268
F.3.	Dibujo de la solución (y de su derivada) del PVI planteado en Ejemplo F.9	276
F.4.	Dibujo de la región $A$ -Estabilidad del método de Adams–Moulton de 3 pasos	280
F.5.	Solución del PVI (F.4).	285
F.6.	Solución del PVI con <code>ode23t</code>	286
F.7.	Población de lince y conejos en un periodo de 10 años	289
F.8.	Ciclo de la vida	290
F.9.	Plano fase en $[0, 10]$	290
F.10.	Plano fase en $[0, 20]$	290
F.11.	Región de $A$ -estabilidad del método de Adams–Bashforth de 4 pasos.	291
F.12.	Solución numérica del PVI usando el método AB de 4 pasos.	292
F.13.	Solución numérica del PVI (F.6).	293
F.14.	Evolución de las poblaciones del PVI (F.8).	297
F.15.	Evolución de las poblaciones con datos diferentes.	297

# Capítulo 3

## DERIVACIÓN NUMÉRICA

**Autores:** Juan F. Mañas–Mañas<sup>1</sup>, Maritza A. Pinta<sup>2</sup>

<sup>1</sup>Departamento de Matemáticas, Universidad de Almería, España.

<sup>2</sup>Unidad Académica de Ingeniería Civil, Universidad Técnica de Machala, Ecuador.

<sup>1</sup>jmm939@ual.es, <sup>2</sup>mpinta@utmachala.edu.ec

El uso de los métodos numéricos para calcular la derivada de una función, se justifica en los casos en que la función a derivar es muy compleja, o, solo se conoce los valores de la función en ciertos puntos. En el resto de casos, basta con utilizar las reglas de derivación analítica, repasadas en el Capítulo 1, ó, software matemáticos como Matlab, Mathematica, Maple ó Maxima (este último gratuito).

El principal objetivo de este capítulo es obtener aproximaciones del valor de la derivada de una función en un punto solamente usando evaluaciones de la función de partida. La principal razón es porque la derivación es un concepto del cálculo infinitesimal, que se define por medio de un proceso de paso al límite. Puesto que este proceso no puede ser reproducido por un ordenador digital, necesitamos sustituirlo por otro proceso finito o de *discretización*.

Para ello usaremos principalmente el desarrollo de Taylor de una función. Otros métodos consistiran en usar el polinomio interpolador de una función y finalmente veremos mejoras de estos métodos como es la Extrapolación de Richardson. El lector podrá encontrar gran variedad de ejemplos resueltos (ver Apéndice C) y todos los códigos asociados a los métodos de derivación numérica son implementados en Matlab.

Para el desarrollo de este capítulo vamos a necesitar un nuevo concepto matemático que es el **Desarrollo de Taylor** ó **Serie de Taylor** ó **Polinomio de Taylor** de una función.

### 3.1. Serie de Taylor

La serie de Taylor nos permite manipular funciones complejas como las logarítmicas, exponenciales y trigonométricas, aproximando el valor de una función en un entorno de un punto  $a$ , en términos del valor de la función y sus derivadas

en dicho punto, basándonos en la teoría de interpolación (ver teoría y prácticas sobre interpolación en [2], [4, Cap. 2], [8, Cap. 1] ó [11, Cap. 8]).

**Definición 3.1.** Llamaremos **Polinomio de Taylor de grado  $n$**  de una función  $f$  suficientemente derivable en el punto  $a$ , y lo denotaremos por  $T_{n,a}[f]$  al polinomio:

$$T_{n,a}[f](x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \dots + \frac{f^{(n)}(a)}{n!}(x-a)^n. \quad (3.1)$$

**Nota 3.1.** Se puede escribir la fórmula (3.1) más compacta de la siguiente forma:

$$T_{n,a}[f](x) = \sum_{i=0}^n \frac{f^{(i)}(a)}{i!}(x-a)^i. \quad (3.2)$$

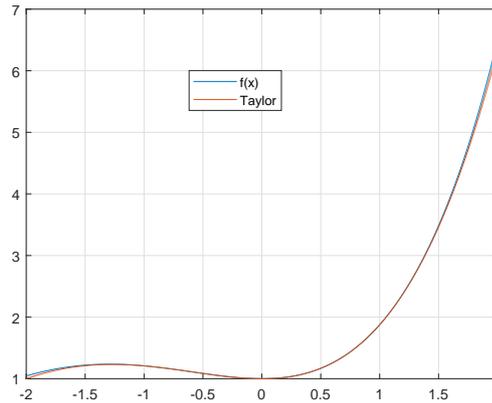
**Teorema 3.1.** [1] Sea “ $f$ ” una función suficientemente derivable en un punto  $a$ , se puede construir el polinomio de Taylor,  $T_{n,a}[f]$ , de tal forma que  $T_{n,a}[f]^{(k)}(a) = f^{(k)}(a)$ , siendo  $k = 1, 2, \dots, n$ , es decir, este polinomio tiene la propiedad de que él y sus derivadas hasta del orden  $n$  coinciden con la función “ $f$ ” y sus derivadas hasta orden  $n$  en el punto  $a$ .

**Ejemplo 3.1.** Calcular el polinomio de Taylor de grado 4 que se ajusta a la función  $f(x) = e^x - \text{sen}(x)$  en el punto  $a = 0$  y estimar el valor en la misma función cuando  $x = 1$ .

**Solución:** El ejercicio pide encontrar la Serie de Taylor de grado 4, para ello es necesario obtener hasta la cuarta derivada y determinar el valor de las mismas en  $a = 0$ :

Función	Valor en $a = 0$
$f(x) = e^x - \text{sen}(x)$	$f(0) = e^0 - \text{sen}(0) = 1 - 0 = 1$
$f'(x) = e^x - \text{cos}(x)$	$f'(0) = e^0 - \text{cos}(0) = 1 - 1 = 0$
$f''(x) = e^x + \text{sen}(x)$	$f''(0) = e^0 + \text{sen}(0) = 1 + 0 = 1$
$f'''(x) = e^x + \text{cos}(x)$	$f'''(0) = e^0 + \text{cos}(0) = 1 + 1 = 2$
$f^{iv}(x) = e^x - \text{sen}(x)$	$f^{IV}(0) = e^0 - \text{sen}(0) = 1 - 0 = 1$

Figura 3.1: Funciones  $f(x)$  y  $T_{4,0}[e^x - \text{sen}(x)](x)$ .



Fuente: Elaboración propia.

Reemplazamos estos valores y  $a = 0$ , en la fórmula (3.2):

$$\begin{aligned}
 T_{4,0}[e^x - \text{sen}(x)](x) &= \sum_{i=0}^4 \frac{f^i(x_0)}{i!} (x - x_0)^i \\
 &= \frac{f(x_0)}{0!} + \frac{f'(x_0)}{1!} (x - x_0) + \frac{f''(x_0)}{2!} (x - x_0)^2 \\
 &\quad + \frac{f'''(x_0)}{3!} (x - x_0)^3 + \frac{f^{IV}(x_0)}{4!} (x - x_0)^4 \\
 &= \frac{f(0)}{0!} + \frac{f'(0)}{1!} x + \frac{f''(0)}{2!} x^2 + \frac{f'''(0)}{3!} x^3 + \frac{f^{IV}(0)}{4!} x^4 \\
 &= 1 + 0(x) + \frac{1}{2}(x)^2 + \frac{2}{6}(x)^3 + \frac{1}{24}(x)^4 \\
 &= 1 + \frac{x^2}{2} + \frac{x^3}{3} + \frac{x^4}{24}.
 \end{aligned}$$

Por lo que en un entorno de 0 podemos aproximar  $f(x)$  por  $T_{4,0}[e^x - \text{sen}(x)](x)$ , veamos como por ejemplo en  $x = 1$  el valor es muy próximo:

$$\begin{aligned}
 f(x) = e^x - \text{sen}(x) &\approx 1 + \frac{x^2}{2} + \frac{x^3}{3} + \frac{x^4}{24}, \\
 f(1) &= 1,876810843651149, \\
 T_{4,0}[e^x - \text{sen}(x)](1) &= 1,875.
 \end{aligned}$$

En la Figura 3.1 se presenta la gráfica de la función  $f(x) = e^x - \text{sen}(x)$  y  $T_{4,0}[e^x - \text{sen}(x)](x)$  para verificar que en un intervalo alrededor del punto  $a = 1$  obtenemos una buena aproximación de la función.

El código de Matlab ha sido (la leyenda se ha añadido desde las opciones de la ventana gráfica de Matlab):

```
xx=(-2:0.001:2);
```

```

yy=exp(xx)-sin(xx);
zz=1+xx.^2/2+xx.^3/3+xx.^4/24;
plot(xx,yy,xx,zz), grid on

```

## 3.2. Error de truncamiento en la serie de Taylor

**Definición 3.2.** [1] El **error de truncamiento** en la Serie de Taylor, producido al finalizar prematuramente el desarrollo de la misma, se puede expresar como el residuo  $R_n(x)$  y viene dado por:

$$\begin{aligned}
 f(x) = & f(a) + f'(a)(x - a) + f''(a)\frac{(x - a)^2}{2!} \\
 & + f'''(a)\frac{(x - a)^3}{3!} + \cdots + f^n(a)\frac{(x - a)^n}{n!} + R_n(x),
 \end{aligned}
 \tag{3.3}$$

donde  $R_n(x)$  en la fórmula (3.3) viene dado por

$$R_n(x) = f^{n+1}(\xi)\frac{(x - a)^{n+1}}{(n + 1)!},
 \tag{3.4}$$

con  $\xi \in (a, x)$ .

## 3.3. Fórmulas de derivación numérica basadas en la Serie de Taylor

Si recordamos, la derivada  $f'(x)$  de la función  $f(x)$ , viene dada por la fórmula vista en el Capítulo 1:

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x + h) - f(x)}{h}.$$

La aproximación numérica más sencilla de la derivada corresponde entonces a la siguiente fórmula, llamada “diferencia finita” [12], donde  $h$  tiene valor fijo y finito, denominado por lo general “**paso**”, la idea intuitiva es muy sencilla, solo es dar pequeños valores de  $h$  y esa será nuestra aproximación,

$$f'(x) \approx \frac{f(x + h) - f(x)}{h},
 \tag{3.5}$$

**Ejemplo 3.2.** Calcular una aproximación de  $f'(0)$  para distintos valores de  $h$ , siendo  $f(x) = e^x - \text{sen}(x)$ .

**Solución:** En este caso sencillo es fácil ver que  $f'(0) = 0$ , veamos las aproximaciones:

Valor de h	Cálculos	Aproximación
$h = 1$	$\frac{f(1+0)-f(0)}{1}$	0,876811
$h = 0,5$	$\frac{f(0,5+0)-f(0)}{0,5}$	0,338591
$h = 0,1$	$\frac{f(0,1+0)-f(0)}{0,1}$	0,053375
$h = 0,01$	$\frac{f(0,01+0)-f(0)}{0,01}$	0,00503338
$h = 0,001$	$\frac{f(0,001+0)-f(0)}{0,001}$	0,000500333

Parece que cuanto más pequeño es  $h$  mejor es la aproximación. Más adelante veremos que esto no es del todo cierto por los errores de cancelación (ver más información sobre errores de cancelación en [8]).  $\square$

Para que esta fórmula (3.5) sea exacta, habría que sumarle el error que conlleva, o sea:

$$f'(x) = \frac{f(x+h) - f(x)}{h} + Error,$$

donde:

$$Error = f'(x) - \frac{f(x+h) - f(x)}{h}.$$

Debido a la imposibilidad de calcular el error con esta fórmula, por los errores de cancelación que pueden darse al restar dos valores muy similares [8] o porque no se conoce el valor real, se han desarrollado a partir de la serie de Taylor, una serie de fórmulas de alta exactitud, llamadas fórmulas de diferencias divididas finitas [12], las cuales demostraremos a continuación:

### 3.3.1. Fórmulas progresivas o de diferencias divididas finitas hacia delante

Cuando no se tienen datos a la izquierda del punto en que se debe calcular la derivada, se utilizan las fórmulas progresivas. Para el desarrollo de las mismas, se considera la función  $f(x)$ , sus puntos,  $x_0$ ,  $x_0 + h$  (punto delante), y, la correspondiente Serie de Taylor, truncada en el tercer término.

**Nota 3.2.** *Truncaremos la Serie de Taylor en el término que nos interese en función de la derivada y el orden que queramos calcular, veremos ejemplos durante el desarrollo del capítulo.*

La fórmula de la Primera diferencia progresiva [7], con error de truncamiento  $O(h)$ , de orden 1 es:

$$f'(x_0) = \frac{f(x_0+h) - f(x_0)}{h} + O(h). \quad (3.6)$$

**Demostración:** Para obtener la fórmula (3.6) solamente es necesario desarrollar la Serie de Taylor en el punto  $a = x_0 + h$  y luego truncar en el término que necesitemos, para ello usamos (3.4):

$$f(x_0 + h) = \frac{f(x_0)}{0!} + \frac{f'(x_0)}{1!}(x_0 + h - x_0) + \underbrace{\frac{f''(x_0)}{2!}(x_0 + h - x_0)^2 + \dots}_{R_n(x) = \frac{f''(\xi)}{2}h^2}$$

$$f(x_0 + h) = f(x_0) + f'(x_0)h + \frac{f''(\xi)}{2}h^2.$$

Despejando  $f'(x_0)$  que es el valor que nos interesa aproximar, se tiene

$$\begin{aligned} f'(x_0) &= \frac{f(x_0 + h)}{h} - \frac{f(x_0)}{h} - \frac{f''(\xi)}{2h}h^2, \\ &= \frac{f(x_0 + h) - f(x_0)}{h} - \underbrace{\frac{f''(\xi)}{2}h}_{\text{Error } O(h)}, \\ &= \frac{f(x_0 + h) - f(x_0)}{h} + O(h), \end{aligned}$$

con  $\xi \in (x_0, x_0 + h)$ .  $\square$

### 3.3.2. Formulas regresivas o de diferencias divididas finitas hacia atrás

Cuando no se tienen datos a la derecha del punto en que se debe calcular la derivada, se utilizan las fórmulas regresivas. Para el desarrollo de las mismas, se considera la función  $f(x)$ , sus puntos,  $x_0$ ,  $x_0 - h$  (punto atrás), y, la correspondiente Serie de Taylor en el punto  $a = x_0 - h$  truncada en el tercer término.

La fórmula de la Primera diferencia regresiva [7], con error de truncamiento  $O(h)$ , de orden 1 es:

$$f'(x_0) = \frac{f(x_0) - f(x_0 - h)}{h} + O(h). \quad (3.7)$$

**Demostración:** La demostración para obtener (3.7) es análoga al caso anterior, hacemos el desarrollo de Taylor en el punto  $a = x_0 - h$ , truncamos y despajamos  $f'(x_0)$  que es lo que queremos aproximar.

$$\begin{aligned}
f(x_0 - h) &= f(x_0) + f'(x_0)(x_0 - h - x_0) + \frac{f''(\xi)}{2}(x_0 - h - x_0)^2 + \dots \\
&= f(x_0) - hf'(x_0) + \frac{f''(\xi)}{2}h^2,
\end{aligned}$$

↓

$$\begin{aligned}
hf'(x_0) &= f(x_0) - f(x_0 - h) + \frac{f''(\xi)}{2}h^2, \\
f'(x_0) &= \frac{f(x_0) - f(x_0 - h)}{h} + \frac{f''(\xi)}{2h}h^2 \\
&= \frac{f(x_0) - f(x_0 - h)}{h} + \underbrace{\frac{f''(\xi)}{2}h}_{\text{Error } O(h)} \\
&= \frac{f(x_0) - f(x_0 - h)}{h} + O(h),
\end{aligned}$$

con  $\xi \in (x_0 - h, x_0)$ . □

**Nota 3.3.** Las fórmulas que hemos obtenido del error son muy importantes, hay que pensar que en casos reales no conocemos el valor exacto de la derivada, por tanto es necesario controlar el error que estamos cometiendo. Este error en las fórmulas de derivación (3.6) y (3.7) viene dado por la expresión

$$\frac{f''(\xi)}{2}h.$$

Si conseguimos acotar  $f''(\xi)$  en el intervalo donde varía  $\xi$  tendremos una cota del error cometido.

Veamos un ejemplo de lo que queremos calcular:

**Ejemplo 3.3.** Una cota del error cometido en el Ejemplo 3.2.

Puesto que en el Ejemplo 3.2 se han usado fórmulas progresivas, se tiene que acotar  $f''(\xi)$  en el intervalo  $(0, 1)$  en el primer caso.

$$f''(x) = e^x + \text{sen}(x).$$

Es fácil ver que  $f''(x)$  en el intervalo  $(0, 1)$  es estrictamente creciente, por tanto el máximo de  $f''(x)$  se encuentra en  $f''(1) = 3,55975$ . Así, el error cometido es menor o igual a

$$\text{Error} \leq \frac{f''(1)}{2} \cdot 1 = 1,77988.$$

Siempre el error real va a ser menor que el error teórico, veamos para el resto de valores de  $h$ :

Valor de h	Cálculos	Error Teórico
$h = 0,5$	$\frac{f''(0,5)}{2} \cdot 0,5$	0,532037
$h = 0,1$	$\frac{f''(0,1)}{2} \cdot 0,1$	0,0602502
$h = 0,01$	$\frac{f''(0,001)}{2} \cdot 0,01$	0,00510025
$h = 0,001$	$\frac{f''(0,0,001)}{2} \cdot 0,001$	0,000501

### 3.3.3. Fórmulas de diferencias finitas centradas

La idea es usar la Serie de Taylor en más de un punto y hacer una combinación lineal de ambas expresiones, en este primer caso tomamos el punto  $x_0$  en el centro, se considera un punto delante,  $x_0 + h$ , y, un punto atrás,  $x_0 - h$ . La fórmula de la primera derivada [7], con error de truncamiento  $O(h^2)$ , de orden 2, es:

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0 - h)}{2h} + O(h^2). \quad (3.8)$$

**Demostración:** Como se ha introducido anteriormente, los pasos para obtener (3.8) son sencillos, primero desarrollamos la Serie de Taylor en los puntos  $a = x_0 + h$  y  $a = x_0 - h$  y luego hacemos una combinación lineal de ambas expresiones. Sea la fórmula adelantada:

$$f(x_0 + h) = f(x_0) + f'(x_0)h + \frac{f''(x_0)h^2}{2!} + \frac{f'''(\xi_1)h^3}{3!}, \quad (3.9)$$

con  $\xi_1 \in (x_0, x_0 + h)$ . Por otro lado, sea la fórmula atrasada:

$$\begin{aligned} f(x_0 - h) &= f(x_0) + f'(x_0)(-h) + \frac{f''(x_0)(-h)^2}{2!} + \frac{f'''(\xi_2)(-h)^3}{3!} + \dots, \\ f(x_0 - h) &= f(x_0) - hf'(x_0) + \frac{h^2 f''(x_0)}{2} - \frac{h^3 f'''(\xi_2)}{6}, \end{aligned} \quad (3.10)$$

con  $\xi_2 \in (x_0 - h, x_0)$ .

Multiplicando por  $-1$  toda la expresión (3.10) y posteriormente sumando la expresión (3.9) obtenemos:

$$\begin{array}{rcll} f(x_0 + h) & = & f(x_0) & + hf'(x_0) & + \frac{h^2}{2} f''(x_0) & + \frac{h^3}{6} f'''(\xi_1) \\ -f(x_0 - h) & = & -f(x_0) & + hf'(x_0) & - \frac{h^2}{2} f''(x_0) & + \frac{h^3}{6} f'''(\xi_2) \\ \hline f(x_0 + h) - f(x_0 - h) & = & & 2hf'(x_0) & & + \frac{h^3}{6} [f'''(\xi_1) + f'''(\xi_2)] \end{array}$$

Así, nos queda que

$$2hf'(x_0) = f(x_0 + h) - f(x_0 - h) - \frac{h^3}{6} [f'''(\xi_1) + f'''(\xi_2)].$$

Por otro lado, aplicando el teorema del valor intermedio es sencillo probar

que:

$$\frac{f'''(\xi_1) + f'''(\xi_2)}{2} = f'''(\xi),$$

con  $\xi \in (x_0 - h, x_0 + h)$ . Finalmente, despejando  $f'(x_0)$  tenemos que:

$$\begin{aligned} f'(x_0) &= \frac{f(x_0 + h) - f(x_0 - h)}{2h} - \frac{h^3 f'''(\xi)}{6h} \\ &= \frac{f(x_0 + h) - f(x_0 - h)}{2h} - \underbrace{\frac{h^2}{6} f'''(\xi)}_{\text{Error } O(h^2)} \\ &= \frac{f(x_0 + h) - f(x_0 - h)}{2h} + O(h^2). \end{aligned}$$

**Nota 3.4.** Es muy importante darse cuenta que para obtener fórmulas de orden superior es necesario realizar combinaciones lineales de manera que al sumar varios desarrollos de Taylor se vayan cancelando las potencias de  $h$ . En otras palabras, la serie de Taylor se puede ver como un polinomio en la variable  $h$ , si nosotros queremos obtener una fórmula para la primera derivada, es cuando  $h$  está elevado a 1, para obtener una fórmula de orden 2 (como la que acabamos de obtener) tenemos que cancelar el factor  $h^2$  y pasar a  $h^3$ , por ello, si miramos los desarrollos (3.9) y (3.10) y especialmente el término  $h^2$  de cada uno de ellos, vemos que son iguales, por eso, al cambiar de signo la expresión (3.10) y sumar, ese término se cancela y pasamos al término  $h^3$ , y nuestra aproximación es de orden 2.

Esto mismo que se ha explicado se puede hacer para obtener fórmulas de mayor orden, o para aproximar distintas derivadas, veremos ejemplos durante el desarrollo del capítulo y en el Apéndice C.

**Ejemplo 3.4.** Obtener una fórmula de derivación centrada de orden 4 [6].

**Solución:** La idea, ya que en el ejercicio se indica que es centrada, es probar en los puntos  $x-2h$ ,  $x-h$ ,  $x+h$  y  $x+2h$ . Para asegurar que es de orden 4 necesitamos llegar hasta el desarrollo de  $h^5$  (ya que la primera derivada tiene término  $h$  y al despejar y dividir perdemos un orden). Con esta información hacemos la Serie de Taylor en cada punto:

$$f(x+2h) = f(x) + 2hf'(x) + 2h^2f''(x) + \frac{4h^3}{3}f'''(x) + \frac{2h^4}{3}f^{(IV)}(x) + \frac{4f^{(V)}(\xi_1)}{15}h^5, \quad (3.11)$$

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(x) + \frac{h^4}{24}f^{(IV)}(x) + \frac{f^{(V)}(\xi_2)}{120}h^5, \quad (3.12)$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f'''(x) + \frac{h^4}{24}f^{(IV)}(x) - \frac{f^{(V)}(\xi_3)}{120}h^5, \quad (3.13)$$

$$f(x-2h) = f(x) - 2hf'(x) + 2h^2f''(x) - \frac{4h^3}{3}f'''(x) + \frac{2h^4}{3}f^{(IV)}(x) - \frac{4f^{(V)}(\xi_4)}{15}h^5. \quad (3.14)$$

donde  $\xi_i$  con  $i \in \{1, 2, 3, 4\}$  pertenece al intervalo asociado a cada desarrollo de Taylor.

En este momento, en el cual tenemos todos los desarrollos, tenemos que obtener una combinación lineal de todas las expresiones desde (3.11) hasta la expresión (3.14), de manera que al hacer la combinación lineal los términos  $h^2$ ,  $h^3$  y  $h^4$  desaparezcan (o análogamente que los términos  $f''(x)$ ,  $f'''(x)$  y  $f^{(IV)}(x)$  desaparezcan).

¿Cómo se puede obtener esta combinación lineal? Pues solamente es necesario escribir este pequeño sistema de ecuaciones **sobredeterminado** (es decir, con más incógnitas que ecuaciones y por tanto siempre tiene infinitas soluciones). Sean  $a$ ,  $b$ ,  $c$  y  $d$  los coeficientes que multipliquen a las expresiones (3.11)-(3.14), respectivamente. Entoces el sistema a resolver es:

$$\begin{cases} 2a + \frac{1}{2}b + \frac{1}{2}c + 2d = 0 \\ \frac{4}{3}a + \frac{1}{6}b - \frac{1}{6}c - \frac{4}{3}d = 0 \\ \frac{2}{3}a + \frac{1}{24}b + \frac{1}{24}c + \frac{2}{3}d = 0 \end{cases}$$

Es importante destacar que la primera ecuación es la formada por los coeficientes del término  $h^2$  en todas las expresiones. Una de las infinitas soluciones puede ser

$$a = 1, \quad b = -8, \quad c = 8 \quad \text{y} \quad d = -1.$$

Ahora, realizando la siguiente combinación lineal:

$$(3.11) - 8 \cdot (3.12) + 8 \cdot (3.13) - (3.14)$$

se obtiene que el término de la izquierda de la igualdad vale

$$f(x + 2h) - 8f(x + h) + 8f(x + h) - f(x + 2h),$$

y por otro lado, los términos de  $h^2$ ,  $h^3$  y  $h^4$  es fácil ver que se han cancelado, y solo nos queda

$$-12hf'(x) + \frac{1}{15}(4f^{(V)}(\xi_1) - f^{(V)}(\xi_2) - f^{(V)}(\xi_3) + 4f^{(V)}(\xi_4))h^5.$$

De donde se obtiene la siguiente fórmula de la primera derivada, con error de cuarto orden:

$$f'(x) = \frac{-f(x + 2h) + 8f(x + h) - 8f(x - h) + f(x - 2h)}{12h} + O(h^4). \quad (3.15)$$

### 3.4. Derivadas de orden superior

De manera análoga, puede utilizarse también la Serie de Taylor, para obtener derivadas de orden superior, como la segunda y tercera derivada, las cuales

sintetizaremos en esta sección.

**Ejemplo 3.5.** *Obten una fórmula para aproximar la  $f''(x)$  con orden 2.*

**Solución:** Desarrollamos la Serie de Taylor en los puntos  $x - h$  y  $x + h$ .

$$\begin{aligned} f(x+h) &= f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(x) + \frac{h^4}{24}f^{(IV)}(\xi_1), \\ f(x-h) &= f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f'''(x) + \frac{h^4}{24}f^{(IV)}(\xi_2), \end{aligned}$$

Sumando ambas expresiones, claramente los términos  $h$  y  $h^3$  se cancelan, por lo que nos queda:

$$\begin{aligned} f(x+h) + f(x-h) &= 2f(x) + f''(x)h^2 + \frac{h^4}{24}(f^{(IV)}(\xi_1) + f^{(IV)}(\xi_2)), \\ f''(x) &= \frac{f(x+h) + f(x-h) - 2f(x)}{h^2} + O(h^2). \end{aligned} \quad (3.16)$$

**Ejemplo 3.6.** *Utilizar las formulas para la primera derivada, y derivadas de orden superior vistas en la Tabla 3.4 para aproximar las primeras derivadas de la función*

$$f(x) = (\sqrt{x} + 2x - 5)^{\ln(x+1)}$$

en el punto  $x_1 = 2$  y con paso  $h = 0,01$ .

**Primera derivada hacia delante.**

$$f'(x_1) \approx \frac{f(x_1+h) - f(x_1)}{h}$$

$$\begin{aligned} f'(2) &\approx \frac{f(2+0,01) - f(2)}{0,01} \\ &= \frac{(\sqrt{2,01} + 2(2,01) - 5)^{\ln(2,01+1)} - (\sqrt{2} + (2(2) - 5)^{\ln(2+1)})}{0,01} \\ &= \frac{(0,437744687)^{\ln(3,01)} - (0,414213562)^{\ln(3)}}{0,01} \\ &= \frac{0,402389753 - 0,379732639}{0,01} = \frac{0,022657114}{0,01} \\ &= \boxed{2,2657114} \end{aligned}$$

Tabla 3.1: Principales fórmulas de derivación numérica

Derivada	Tipo de fórmula	Fórmula	Orden del error
$f''(x)$	Hacia Adelante	$\frac{f(x+2h)-2f(x+h)+f(x)}{h^2}$	$O(h)$
		$\frac{-f(x+3h)+4f(x+2h)-5f(x+h)+2f(x)}{h^2}$	$O(h^2)$
	Centrada	$\frac{f(x+h)-2f(x)+f(x-h)}{h^2}$	$O(h^2)$
		$\frac{-f(x+2h)+16f(x+h)-30f(x)+16f(x-h)-f(x-2h)}{12h^2}$	$O(h^4)$
	Hacia atrás	$\frac{f(x)-2f(x+h)+f(x-2h)}{h^2}$	$O(h)$
		$\frac{2f(x)-5f(x-h)+4f(x-2h)-f(x-3h)}{h^2}$	$O(h^2)$
$f'''(x)$	Hacia Adelante	$\frac{f(x+3h)-3f(x+2h)+3f(x+h)-f(x)}{h^3}$	$O(h)$
		$\frac{-3f(x+4h)+14f(x+3h)-24f(x+2h)+18f(x+h)-5f(x)}{2h^3}$	$O(h^2)$
	Centrada	$\frac{f(x+h)-2f(x+h)+2f(x-h)-f(x-2h)}{2h^3}$	$O(h^2)$
		$\frac{-f(x+3h)+8f(x+2h)-13f(x+h)+13f(x-h)-8f(x-2h)+f(x-3h)}{8h^3}$	$O(h^4)$
	Hacia atrás	$\frac{f(x)-3f(x-h)+3f(x-2h)-f(x-3h)}{h^3}$	$O(h)$
		$\frac{5f(x)-18f(x-h)+24f(x-2h)-14f(x-3h)+3f(x-4h)}{2h^3}$	$O(h^2)$

Fuente: Elaboración propia

### Primera derivada hacia atrás

$$f'(x_1) \approx \frac{f(x_1) - f(x_1 - h)}{h}$$

$$\begin{aligned} f'(2) &\approx \frac{f(2) - f(2 - 0,01)}{0,01} \\ &= \frac{(\sqrt{2} + 2(2) - 5)^{\ln(2+1)} - (\sqrt{1,99} + 2(1,99) - 5)^{\ln(1,99+1)}}{0,01} \\ &= \frac{0,379732639 - 0,357210984}{0,01} = \frac{0,022521654}{0,01} \\ &= \boxed{2,252165452} \end{aligned}$$

### Primera derivada central

$$f'(x_1) \approx \frac{f(x_1 + h) - f(x_1 - h)}{2h}$$

$$\begin{aligned} f'(2) &\approx \frac{f(2 + 0,01) - f(2 - 0,01)}{2(0,01)} \\ &= \frac{(\sqrt{2,01} + 2(2,01) - 5)^{\ln(2,01+1)} - (\sqrt{1,99} + 2(1,99) - 5)^{\ln(1,99+1)}}{2(0,01)} \\ &= \frac{0,402389753 - 0,357210984}{0,02} = \frac{0,045178769}{0,02} \\ &= \boxed{2,25893845} \end{aligned}$$

### Segunda derivada hacia delante

$$f''(x_1) \approx \frac{f(x_1 + 2h) - 2f(x_1 + h) + f(x_1)}{h^2}$$

$$\begin{aligned} f''(2) &\approx \frac{f(2 + 2(0,01)) - 2f(2 + 0,01) + f(2)}{(0,01)^2} \\ &= \frac{(\sqrt{2,02} + 2(2,02) - 5)^{\ln(2,02+1)} - 2(\sqrt{2,01} + 2(2,01) - 5)^{\ln(2,01)+1}}{(0,01)^2} \\ &\quad + \frac{(\sqrt{2} + 2(2) - 5)^{\ln(2+1)}}{(0,01)^2} \\ &= \frac{0,425188118 - 2(0,402389753) + 0,379732639}{0,0001} = \frac{0,000141251}{0,0001} \\ &= \boxed{1,41251} \end{aligned}$$

### Segunda derivada hacia atrás

$$f''(x_1) \approx \frac{f(x_1) - 2f(x_1 - h) + f(x_1 - 2h)}{(h)^2}$$

$$\begin{aligned} f''(2) &\approx \frac{f(2) - 2f(2 - 0,01) + f(2 - 2(0,01))}{(0,01)^2} \\ &= \frac{(\sqrt{2} + 2(2) - 5)^{\ln(2+1)} - 2(\sqrt{1,99} + 2(1,99) - 5)^{\ln(1,99+1)}}{0,0001} \\ &\quad + \frac{(\sqrt{1,98} + 2(1,98) - 5)^{\ln(1,98+1)}}{0,0001} \\ &= \frac{0,379732639 - 2(0,357210984) + 0,334818827}{0,0001} = \frac{0,000129498}{0,0001} \\ &= \boxed{1,294982895} \end{aligned}$$

### Segunda derivada centrada

$$f''(x_1) \approx \frac{f(x_1 + h) - 2f(x_1) + f(x_1 - h)}{h^2}$$

$$\begin{aligned} f''(2) &\approx \frac{f(2 + 0,01) - 2f(2) + f(2 - 0,01)}{(0,01)^2} \\ &= \frac{(\sqrt{2,01} + 2(2,01) - 5)^{\ln(2,01+1)} - 2(\sqrt{2} + 2(2) - 5)^{\ln(2+1)}}{(0,01)^2} \\ &\quad + \frac{(\sqrt{1,99} + 2(1,99) - 5)^{\ln(1,99)+1}}{0,0001} \\ &= \frac{0,402389753 - 2(0,379732639) + 0,357210984}{0,0001} = \frac{0,000135459}{0,0001} \\ &= \boxed{1,35459} \end{aligned}$$

### Tercera derivada hacia delante

$$f'''(x_1) \approx \frac{f(x_1 + 3h) - 3f(x_1 + 2h) + 3f(x_1 + h) - f(x_1)}{h^3}$$

$$\begin{aligned}
f'''(2) &\approx \frac{f(2 + 3(0,01)) - 3f(2 + 2(0,01)) + 3f(2 + 0,01) - f(2)}{(0,01)^3} \\
&= \frac{(\sqrt{2,03} + 2(2,03) - 5)^{\ln(2,03+1)} - 3(\sqrt{2,02} + 2(2,02) - 5)^{\ln(2,02-1)}}{(0,01)^3} \\
&\quad + \frac{3(\sqrt{2,01} + 2(2,02) - 5)^{\ln(2,01+1)} - (\sqrt{2} + 2(2) - 5)^{\ln(2+1)}}{(0,01)^3} \\
&= \frac{0,448133368 - 3(0,425188118) + 3(0,402389753) - 0,379732639}{0,000001} \\
&= \frac{0,000005634}{0,000001} = \boxed{5,634}
\end{aligned}$$

**Tercera derivada hacia atrás**

$$f'''(x_1) \approx \frac{f(x_1) - 3f(x_1 - h) + 3f(x_1 - 2h) - f(x_1 - 3h)}{(h)^3}$$

$$\begin{aligned}
f'''(2) &\approx \frac{f(2) - 3f(2 - 0,01) + 3f(2 - 2(0,01)) - f(2 - 3(0,01))}{(0,01)^3} \\
&= \boxed{6,133343368}
\end{aligned}$$

**Tercera derivada centrada**

$$f'''(x_1) \approx \frac{f(x_1 + 2h) - 2f(x_1 + h) + 2f(x_1 - h) - f(x_1 - 2h)}{2h^3}$$

$$\begin{aligned}
f'''(2) &= \frac{f(2 + 2(0,01)) - 2f(2 + 0,01) + 2f(2 - 0,01) - f(2 - 2(0,01))}{2h^3} \\
&= \boxed{5,8765}
\end{aligned}$$

**Cuarta derivada hacia delante**

$$f^{(IV)}(x_1) \approx \frac{f(x_1 + 4h) - 4f(x_1 + 3h) + 6f(x_1 + 2h) - 4f(x_1 + h) + f(x_1)}{h^4}$$

$$\begin{aligned}
f^{(IV)}(2) &\approx \frac{f(2 + 4(0,01)) - 4f(2 + 3(0,01)) + 6f(2 + 2(0,01))}{(0,01)^4} \\
&\quad - \frac{4f(2 + 0,01) + f(2)}{(0,01)^4} \\
&= \boxed{-15,5}
\end{aligned}$$

**Cuarta derivada hacia atrás**

$$\boxed{f^{(IV)}(x_1) \approx \frac{f(x_1) - 4f(x_1 - h) + 6f(x_1 - 2h) - 4f(x_1 - 3h) + f(x_1 - 4h)}{h^4}}$$

$$\begin{aligned}
f^{(IV)}(2) &\approx \frac{f(2) - 4f(2 - 0,01) + 6f(2 - 2(0,01)) - 4f(2 - 3(0,01))}{(0,01)^4} \\
&\quad + \frac{f(2 - 4(0,01))}{(0,01)^4} \\
&= \boxed{-17,2756033}
\end{aligned}$$

**Cuarta derivada centrada**

$$\boxed{f^{(IV)}(x_1) \approx \frac{f(x_1 + 2h) - 4f(x_1 + h) + 6f(x_1) - 4f(x_1 - h) + f(x_1 - 2h)}{h^4}}$$

$$\begin{aligned}
f^{(IV)}(2) &= \frac{f(2 + 2(0,01)) - 4f(2 + 0,01) + 6f(2) - 4f(2 - 0,01)}{(0,01)^4} \\
&\quad + \frac{f(2 - 2(0,01))}{(0,01)^4} \\
&= \boxed{-16.9}
\end{aligned}$$

(Ver codificación en Matlab en Apéndice C)

### 3.5. Fórmulas basadas en polinomios de interpolación

Estas fórmulas sirven tanto si los datos están equiespaciados, o no, y, utilizan entre otros, los polinomios de interpolación de Lagrange y Newton. La idea es muy sencilla, sabemos que un polinomio es muy facil de derivar, por lo tanto, si tenemos un polinomio que aproxima a la función  $f(x)$  (ver capítulos de interpolación en

[2], [4, Cap. 2], [8, Cap. 1] ó [11, Cap. 8]) tenemos que

$$f(x) = p(x) + e(f(x)),$$

por lo tanto,

$$f^{(k)}(x) = p^{(k)}(x) + e^{(k)}(f(x)), \quad \forall k = 0, 1, 2, \dots$$

y vamos a tomar como aproximación

$$f^{(k)}(x) \approx p^{(k)}(x), \quad \forall k = 0, 1, 2, \dots$$

Con esta sencilla idea vamos a *redescubrir* algunas de las fórmulas de derivación más utilizadas (muchas de ellas son las anteriores).

### 3.5.1. Utilizando el polinomio interpolador de Lagrange

Dada una función  $f(x)$ , podemos obtener sus fórmulas de derivación, construyendo su polinomio de interpolación, y, tomando sus derivadas como aproximaciones a las derivadas de la función original [13]. Así:

$$f(x) = p_n(x) + e(f(x)) = \sum_{i=0}^n f(x_i) \ell_i(x) + \frac{f^{(n+1)}(\xi)}{(n+1)!} w(x), \quad (3.17)$$

donde  $p_n(x)$  es el polinomio interpolador de Lagrange, que viene dado por

$$p_n(x) = \sum_{i=0}^n f(x_i) \ell_i(x),$$

y los polinomios  $\ell_i(x)$  son los polinomios básicos de Lagrange que están definidos por

$$\ell_i(t) = \prod_{\substack{k=0 \\ k \neq i}}^n \left( \frac{x - x_k}{x_i - x_k} \right).$$

Por otro lado, si miramos cualquier libro de la bibliografía sobre interpolación, sabemos que el error de interpolación que cometemos viene dado por la fórmula de Cauchy, esto es:

$$e(f(x)) = \frac{f^{(n+1)}(\xi)}{(n+1)!} w(x),$$

con  $w(x)$  el polinomio nodal dado por  $w(x) = (x - x_0)(x - x_1) \dots (x - x_n)$ .

Derivando la expresión (3.17) tenemos:

$$f'(x) = \sum_{i=0}^n f(x_i) \ell'_i(x) + \frac{1}{(n+1)!} \left( f^{(n+1)}(\xi) w'(x) + w(x) \frac{\partial f^{(n+1)}(\xi)}{\partial x} \right),$$

Particularizando para  $x = x_k$  con  $k = 0, 1, \dots, n$ ,

$$f'(x_k) = \sum_{i=0}^n f(x_i) \ell'_i(x_k) + \frac{1}{(n+1)!} \left( f^{(n+1)}(\xi) \prod_{j=0, j \neq k}^n (x_k - x_j) \right).$$

### Obtención de algunas fórmulas de derivación numérica usando el polinomio interpolador de Lagrange

Utilizando el polinomio de Lagrange, vamos a deducir las fórmulas para la primera y segunda derivada. Usamos tres puntos:  $x_0 = x-h$ ,  $x_1 = x$  y  $x_2 = x+h$ .

$$\begin{aligned} P_2(t) &= f(x_0)\ell_0(t) + f(x_1)\ell_1(t) + f(x_2)\ell_2(t) \\ &= f(x_0) \frac{(t-x_1)(t-x_2)}{(x_0-x_1)(x_0-x_2)} + f(x_1) \frac{(t-x_0)(t-x_2)}{(x_1-x_0)(x_1-x_2)} \\ &\quad + f(x_2) \frac{(t-x_0)(t-x_1)}{(x_2-x_0)(x_2-x_1)} \\ &= f(x-h) \frac{(t-x)(t-x-h)}{(x-h-x)(x-h-x-h)} + f(x) \frac{(t-x+h)(t-x-h)}{(x-x+h)(x-x-h)} \\ &\quad + f(x+h) \frac{(t-x+h)(t-x)}{(x+h-x+h)(x+h-x)} \\ &= f(x-h) \frac{(t-x)(t-x-h)}{2h^2} - f(x) \frac{(t-x+h)(t-x-h)}{h^2} \\ &\quad + f(x+h) \frac{(t-x+h)(t-x)}{2h^2}, \end{aligned}$$

Ahora, derivando en la expresión obtenida respecto de la variable  $t$  y posteriormente evaluando en  $x$ , que es donde queremos obtener la aproximación de la derivada, tenemos:

$$\begin{aligned} P'_2(t) &= f(x-h) \frac{2t-2x-h}{2h^2} - f(x) \frac{2t-2x}{h^2} + f(x+h) \frac{2t-2x+h}{2h^2}, \\ P'_2(x) &= f(x-h) \frac{2x-2x-h}{2h^2} - f(x) \frac{2x-2x}{h^2} + f(x+h) \frac{2x-2x+h}{2h^2}, \\ P'_2(x) &= -\frac{f(x-h)}{2h} + \frac{f(x+h)}{2h}. \end{aligned}$$

Por lo tanto, hemos obtenido que

$$P'_2(x) = \frac{f(x+h) - f(x-h)}{2h}, \quad (3.18)$$

y podemos aproximar

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}.$$

**Nota 3.5.** Hemos recuperado la fórmula de derivación por el método de Tay-

lor (3.8), lo siguiente que vamos a hacer es derivar dos veces  $P_2(t)$  y vamos a recuperar la fórmula para la segunda derivada de la página 64.

Derivamos dos veces  $P_2(x)$  y evaluamos en  $x$  para obtener:

$$\begin{aligned} P_2''(t) &= f(t-h) \frac{2(1)-0}{2h^2} - f(t) \frac{2(1)-0}{h^2} + f(t+h) \frac{2(1)-0}{2h^2}, \\ P_2''(x) &= \frac{f(x-h)}{h^2} - \frac{2f(x)}{h^2} + \frac{f(x+h)}{h^2} \\ &= \frac{f(x-h)}{h^2} - \frac{2f(x)}{h^2} + \frac{f(x+h)}{h^2}. \end{aligned}$$

Con ello tenemos:

$$f''(t) \approx \frac{f(x-h) - 2f(x) + f(x+h)}{h^2}. \quad (3.19)$$

### 3.5.2. El polinomio interpolador de Newton

Vamos a deducir, manera de ejemplo, las fórmulas de la primera derivada, utilizando dos y tres puntos, y, el polinomio interpolador de Newton [8].

**Fórmula con dos puntos:**

Sean los puntos  $x_0$  y  $x_1$ :

$$p(x) = f(x_0) + f[x_0, x_1](x - x_0), \quad (3.20)$$

donde  $f[x_0, x_1]$  es la diferencia dividida de Newton definida por

$$\begin{aligned} f[x_k] &= f(x_k), \\ f[x_k, x_{k+1}] &= \frac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k}, \\ f[x_k, x_{k+1}, x_{k+2}] &= \frac{f[x_{k+1}, x_{k+2}] - f[x_k, x_{k+1}]}{x_{k+2} - x_k}, \\ &\vdots \\ f[x_k, x_{k+1}, \dots, x_{k+i}] &= \frac{f[x_{k+1}, x_{k+2}, \dots, x_{k+i}] - f[x_k, x_{k+1}, \dots, x_{k+i-1}]}{x_{k+i} - x_k}. \end{aligned}$$

Derivando (3.20) nos queda:

$$p'(x) = f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0}.$$

Si hacemos  $x_1 = x_0 + h$ , al reemplazar en la ecuación anterior, nos quedaría la fórmula (3.6) de la primera derivada, obtenida mediante la serie de Taylor.

**Fórmula con tres puntos:**

Sean los puntos  $x_0$ ,  $x_1$  y  $x_2$ , su polinomio interpolador es:

$$p(x) = f(x_0) + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1).$$

Derivando:

$$p'(x) = f[x_0, x_1] + f[x_0, x_1, x_2](2x - (x_0 + x_1))',$$

finalmente, evaluando los puntos  $x_0 = x - h$ ,  $x_1 = x$  y  $x_2 = x + h$  obtenemos la fórmula de derivación centrada y de orden 2 para la primera derivada.

*Se deja al lector como ejercicio obtener nuevas fórmulas usando distintos polinomios interpoladores.*

### 3.6. Inestabilidad de las fórmulas de derivación numérica

En las fórmulas de derivación numérica antes vistas, se cometen errores. Así por ejemplo, considerando la formula progresiva para la primera derivada, al calcular los valores de  $f(x + h)$  y  $f(x)$ , se cometen errores de redondeo:  $e(h)$  y  $e(-h)$ , respectivamente [13].

Para ejemplificarlo, vamos a seguir el contenido y el ejemplo de [8, Pags. 160-162]. Sabemos que para la primera derivada tenemos

$$f'(x) = D_h(f) - \frac{f''(\xi)}{2}h, \quad D_h(f) = \frac{f(x+h) - f(x)}{h}.$$

Así, si existe  $|f''(\xi)| \leq M_2$ , entonces teóricamente el error vendría dado por

$$|f'(x) - D_h(f)| \leq \frac{M_2}{2}h,$$

lo que nos lleva a pensar que *mientras menor sea el paso  $h$ , menor es el error*, llegando a pensar que el error será 0 cuando  $h \rightarrow 0$ .

Pero no podemos evitar errores de cancelación, por lo que no evaluamos la función  $f$ , sino una perturbación dada por  $f(x) = \tilde{f}(x) + e(x)$ . Usando la linealidad del operador derivada  $D_h(f)$  tendríamos:

$$f'(x) - D_h(\tilde{f}) = f'(x) - D_h(f) - D_h(e).$$

Si asumimos que  $\max\{e(x) : x \in [x, x+h]\} := \|e\|_\infty \leq \varepsilon$ , entonces

$$|D_h(e)| \leq \left| \frac{e(x+h) - e(x)}{h} \right| \leq \frac{2\varepsilon}{h}.$$

De esta manera, la cota real sería:

$$|f'(x) - D_h(\tilde{f})| \leq \frac{M_2}{2}h + \frac{2\varepsilon}{h}.$$

Para valores *muy pequeños* de  $h$ , tenemos un valor  $O(1/h)$  que domina, por lo tanto, hay un valor mínimo de  $h$  que es el error más pequeño al que podemos aspirar. Vamos a ver esto con un ejemplo (ver [8, Ej. 1.4]).

**Ejemplo 3.7.** *Vamos a comprobar las conclusiones anteriores por medio del siguiente experimento. Sea  $f(x) = \sin(x)$  y vamos a aproximar  $f'(\pi/3,2) = \cos(\pi/3,2)$  usando la derivada progresiva vista en (3.6). Vamos a usar los pasos  $h = 10^{-4}, 10^{-5}, \dots, 10^{-12}$ .*

**Solución:** Vamos a usar Matlab, con este pequeño código Matlab nos devuelve todos los valores requeridos:

```
format long
p=(4:12);
paso=10.^(-p);
aprox=(sin(pi/3.2+paso)-sin(pi/3.2))./paso;
error=aprox-cos(pi/3.2)
```

Los resultados obtenidos son:

<b>h</b>	<b>Error cometido</b>
$10^{-4}$	-0,0000415744072991275
$10^{-5}$	-0,0000041573645016557
$10^{-6}$	-0,0000004157574774455
$10^{-7}$	-0,0000000420564073567
$10^{-8}$	0,0000000012422906037
$10^{-9}$	0,0000000234467510962
$10^{-10}$	-0,0000001985978538288
$10^{-11}$	-0,0000035292669278153
$10^{-12}$	-0,0000146314971739558

Como vemos, el error más pequeño es cuando  $h = 10^{-8}$ , lo que coincide con el análisis anterior.  $\square$

### 3.7. Extrapolación de Richardson

La extrapolación de Richardson, es un método de derivación numérica que busca mejorar la precisión del cálculo de la derivada, disminuyendo el error limitado por el parámetro  $h$  [9]. Si pensamos en los métodos obtenidos por el desarrollo de Taylor, al truncar la serie tenemos un error que viene dado por

$$e(f) = c_0h + c_1h^{n+1} + c_2h^{n+2} + \dots$$

¿Podríamos modificar el método numérico para pasar de un  $O(h^n)$  a un paso  $O(h^{n+1})$ ?, en otras palabras, ¿podemos cancelar el término  $O(h^n)$ ? **SI**, para verlo se hará lo siguiente: Considérese la función  $N(h)$ :

$$N(h) = N_0 + \alpha h^m + O(h^{m+1}), \quad (3.21)$$

Siendo  $N(h)$  una aproximación a  $N_0$  de orden  $m$  y  $h \rightarrow 0$ . Sustituyendo  $h$  por  $\tau h$ , siendo  $0 < \tau$  y  $\tau \neq 1$ , nos queda:

$$N(\tau h) = N_0 + \alpha h^m \tau^m + O(h^{m+1}), \quad (3.22)$$

Multiplicando por  $\tau^m$  la expresión (3.21) y restándole la (3.22), tenemos:

$$\begin{aligned} \tau^m N(h) &= \tau^m N_0 + \tau^m \alpha h^m + \tau^m O(h^{m+1}) \\ \frac{-N(\tau h)}{\tau^m N(h) - N(\tau h)} &= \frac{-N_0 - \tau^m \alpha h^m - O(h^{m+1})}{\tau^m N_0 - N_0 + \tau^m O(h^{m+1}) - O(h^{m+1})} \end{aligned}$$

Con esto, obtenemos que:

$$\begin{aligned} \tau^m N(h) - N(\tau h) &= (\tau^m - 1)N_0 + (\tau^m - 1)O(h^{m+1}) \\ \tau^m N(h) - N(\tau h) &= (\tau^m - 1)[N_0 + O(h^{m+1})] \\ \frac{\tau^m N(h) - N(\tau h)}{\tau^m - 1} &= N_0 + O(h^{m+1}) \end{aligned}$$

¿Qué acabamos de obtener? Pues hemos obtenido un método numérico

$$\frac{\tau^m N(h) - N(\tau h)}{\tau^m - 1}$$

que converge a una cantidad  $N_0$  con un orden  $O(h^{n+1})$ . Es importante fijarnos que este proceso se puede repetir y pasar a obtener otro nuevo método numérico de orden  $O(h^{n+2})$ ,  $O(h^{n+3})$ , etc... En realidad, acabamos de probar el siguiente teorema:

**Teorema 3.2 (Extrapolación de Richardson reiterada).** [10] *Supongamos que para la función  $N(h)$  es válido el desarrollo asintótico*

$$N(h) = N_0 + n_1 h^{p_1} + n_2 h^{p_2} + \dots, \quad 0 < p_1 < p_2 \dots, \quad h \rightarrow 0.$$

Sean  $\tau > 0$  y  $\tau \neq 1$ , y la sucesión  $N_n(h)$ , dada recursivamente por:

$$\begin{aligned} N_1(h) &= N(h), \\ N_{n+1}(h) &= \frac{\tau^{p_n} N_n(h) - N_n(\tau h)}{\tau^{p_n} - 1}, \quad n \in \mathbb{N}. \end{aligned} \quad (3.23)$$

Entonces

$$N_n(h) = N_0 + O(h^{p_n}), \quad h \rightarrow 0.$$

**Nota 3.6.** *Es muy importante destacar que este método que acabamos de desarrollar es válido para cualquier fórmula de derivación numérica, incluso en el capítulo siguiente, lo podremos usar para integración numérica.*

En el Apéndice C veremos su implementación en Matlab y más ejemplos, pero ahora vamos a ver este método usando los datos del Ejemplo 3.7 y realizandolo a mano:

**Ejemplo 3.8.** Usando los datos obtenidos en el Ejemplo 3.7, para  $h = 10^{-4}$ ,  $h = 10^{-5}$  y  $h = 10^{-6}$  que son respectivamente 0,555528658612303, 0,555566075655101 y 0,555569817262125. Aplicar la fórmula (3.23) con  $\tau = 10$  y obtener una mejor aproximación de  $\cos(\pi/3,2)$ .

**Solución:** Claramente usando el Teorema de la Extrapolación de Richardson, rapidamente tenemos:

$$\begin{aligned} N_1(10^{-4}) &= 0,555528658612303, \\ N_1(10^{-5}) &= 0,555566075655101, \\ N_1(10^{-6}) &= 0,555569817262125. \end{aligned}$$

Ahora necesitamos calcular  $N_2(h)$ , para ello usamos (3.23):

$$\begin{aligned} N_2(10^{-5}) &= \frac{10 \times N_1(10^{-5}) - N_1(10 \times 10^{-5})}{10 - 1} = 0,555570233104301, \\ N_2(10^{-6}) &= \frac{10 \times N_1(10^{-6}) - N_1(10^{-5})}{10 - 1} = 0,555570232996239. \end{aligned}$$

Por último,

$$N_3(10^{-6}) = \frac{10^2 \times N_2(10^{-6}) - N_2(10^{-5})}{100 - 1} = 0,555570232995148.$$

Por tanto, la mejor aproximación es  $N_3(10^{-6}) = 0,555570232995148$ . Los datos se pueden escribir en forma de matriz triangular inferior, y esto nos va a ayudar a la hora de implementarlo en Matlab de forma eficiente (ver implementación en Matlab en Apéndice C):

$$\begin{aligned} N_1(10^{-4}) &= 0,555528658612303 \\ N_1(10^{-5}) &= 0,555566075655101 & N_2(10^{-5}) &= 0,555570233104301 \\ N_1(10^{-6}) &= 0,555569817262125 & N_2(10^{-6}) &= 0,555570232996239 & N_3(10^{-6}) &= 0,555570232995148 \end{aligned}$$

**Nota 3.7.** La forma de llenar esta matriz es muy sencilla, la primera columna es aplicar el método numérico que queramos, por ejemplo, la derivada progresiva con distintos valores de  $h$ , en realidad, lo valores de  $h$  son  $h/\tau^{i-1}$  donde  $i$  indica la posición o fila de la matriz, es decir, eligiendo el  $h$  de partida en la posición de la matriz  $a_{1,1}$ , en este caso  $h = 10^{-4}$ , el resto vienen prefijados.

Para el resto de columnas, la posición  $a_{i,j}$  con  $j \leq 2$ , se llenan usando unicamente  $a_{i,j-1}$  y  $a_{i-1,j-1}$  y la fórmula (3.23).

**Nota 3.8.** En la práctica se suele usar  $\tau = 2$  por estabilidad computacional (ver [8, Cap. 3]).



# Bibliografía

- [1] Bartle, R. (2004) *Introducción al Análisis Matemático de una variable*. México D. F. México: Limusa S.A.
- [2] Chapra, S. C. (2011) *Métodos Numéricos para ingenieros*. Mexico D. F., México: Editorial McGraw Hill.
- [3] García, J. (23 de julio de 2017) *Interpolación y Aproximación, Integración*. Ediciones Universidad de Lleida. Obtenido de [http://numat.net/mat/an\\_vol\\_1.pdf](http://numat.net/mat/an_vol_1.pdf)
- [4] Gautschi, W. (2002) *Numerical Analysis*. New York, Estados Unidos: Springer.
- [5] Infante del Río, J. y Rey-Cabezas, J. M. (2015) *Métodos numéricos: teoría, problemas y prácticas con MATLAB (4a. ed.)*. Madrid, España: Ediciones Pirámide ProQuest ebrary.
- [6] Issac A. y García, S. M. (2009) *Métodos numéricos: problemas resueltos y prácticas*. Lleida, España: Ediciones de la Universidad de Lleida.
- [7] Jerves, A. (2008) *Elementos de cálculo numérico*. Cuenca, Ecuador: Ediciones Abya-Yala.
- [8] Martínez-Finkelshtein, A. y Moreno-Balcázar, J. J. (1999) *Métodos Numéricos: Aproximación en  $\mathbb{R}$* . Almería, España: Servicio de Publicaciones de la Universidad de Almería.
- [9] Moreno, C. (2007) *Introducción al cálculo numérico*. Madrid, España: Universidad Nacional de Educación a distancia.
- [10] Moreno-Balcázar, J. J. (2017) *Apuntes de Métodos Numéricos II*. Almería, España: Grado en Matemáticas de la Universidad de Almería.
- [11] Quarteroni, A. Sacco, R. y Saleri, F. (2000) *Numerical Mathematics*. New York, Estados Unidos: Springer.
- [12] Vázquez, L. y Jiménez, S. (2009) *Métodos numéricos para la física y la ingeniería*. Madrid, España: Editorial McGraw-Hill España.
- [13] Vigo Aguilar, J. R. (22 de julio de 2016) *Utilización de nuevas tecnologías en la asignatura de Análisis Numérico en el grado de Matemáticas*.



# Apéndice C

En este Apéndice se encuentran las implementaciones (sintaxis) en Matlab de ejercicios referentes al capítulo de Derivación Numérica. Además, se encuentran ejemplos resueltos extras del contenido del capítulo.

## C.1. Ejemplo de la Serie de Taylor.

**Ejemplo C.1.** Sea  $f(x) = 4x \cdot \text{sen}(x)$ , hallar su aproximación usando la serie de Taylor de tercer orden con  $a = \frac{\pi}{2}$ . Predecir el valor de  $f(x)$  en  $x = 1$ , encontrar el error de truncamiento y dibujar en un entorno de  $a = \frac{\pi}{2}$  las funciones  $f(x)$  y  $T_{3,\pi/2}[4x \cdot \text{sen}(x)](x)$ .

**Solución:** Lo primero es construir el polinomio de Taylor desarrollado en el punto  $a = \frac{\pi}{2}$ .

$$\begin{aligned} T_{3,\pi/2}[4x \cdot \text{sen}(x)](x) &= \sum_{i=0}^3 \frac{f^{(i)}(x)}{i!} \left(x - \frac{\pi}{2}\right)^i \\ &= 2\pi + 4 \left(x - \frac{\pi}{2}\right) - \pi \left(x - \frac{\pi}{2}\right)^2 - 2 \left(x - \frac{\pi}{2}\right)^3 \\ &= \left(4 - \frac{\pi^2}{2}\right)x + 2\pi x^2 - 2x^3. \end{aligned}$$

Ahora toca predecir el valor en  $x = 1$ ,

$$T_{3,\pi/2}[4x \cdot \text{sen}(x)](1) = \left(4 - \frac{\pi^2}{2}\right) + 2\pi - 2 = 3,34838.$$

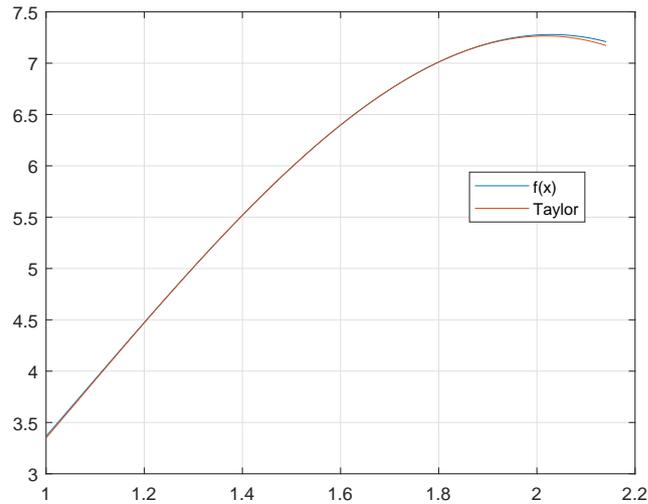
Por otro lado, si calculamos el valor real, sería  $f(1) = 3,36588$ , vemos que la aproximación es bastante buena.

El error de truncamiento, puesto que conocemos el valor real sería

$$E_T = |3,34838 - 3,36588| = 0,0175008.$$

Si no conocemos el valor real tenemos que acotar  $f^{(IV)}(x)$  en el intervalo  $[1, \pi - 1]$ .

Figura C.1: Funciones  $f(x)$  y  $T_{3,\pi/2}[4x \cdot \text{sen}(x)](x)$ .



Fuente: Elaboración propia.

$$f^{(IV)}(x) = 4x \cdot \text{sen}(x) - 16 \cos(x) \quad \Rightarrow \quad |f^{(IV)}(x)| \leq 4(\pi - 1) + 16 = 24,5664,$$

así una cota del error viene dado por

$$E_T \leq \frac{f^{(IV)}(\xi)}{4!} \left(1 - \frac{\pi}{2}\right)^4 = 0,108656.$$

Por último, dibujamos las gráficas solicitadas se pueden ver en la Figura C.1.  $\square$

## C.2. Grado de exactitud

**Definición C.1.** Dada una fórmula de aproximación para la derivada  $k$ -ésima en el punto  $a$  de la forma

$$D_f^{(k)}(a) = \sum_{i=0}^n \alpha_i f(x_i),$$

diremos que el máximo valor  $m$  tal que una fórmula aproximada como las anteriores es exacta para todo  $f \in \mathbb{P}_m$  se llama **grado de exactitud** de dicha fórmula.

**Ejemplo C.2.** Hallar los coeficientes  $a_i$  para que la siguiente fórmulas de derivación numérica sea exacta en  $\mathbb{P}_2$ :

$$f'(2) \approx a_1 f(0) + a_2 f(2) + a_3 f(4).$$

**Solución:** Para que una fórmula sea exacta en  $\mathbb{P}_2$ , es necesario solamente justificar que es exacta en una base de  $\mathbb{P}_2$ , o lo que es lo mismo, justificar la exactitud en este caso en la base canónica  $\{1, t, t^2\}$ .

- Veamos la exactitud para  $f(t) = 1$ . Por un lado derivamos y por otro lado aplicamos la fórmula de derivación, y luego igualamos:

$$f(t) = 1 \Rightarrow f'(t) = 0 \Rightarrow f'(2) = 0.$$

$$D_f^{(1)}(2) = a_1 f(0) + a_2 f(2) + a_3 f(4) = a_1 \cdot 1 + a_2 \cdot 1 + a_3 \cdot 1 = a_1 + a_2 + a_3.$$

Por lo tanto, nuestra primera ecuación sería

$$0 = a_1 + a_2 + a_3.$$

- Veamos la exactitud para  $f(t) = t$ . Por un lado derivamos y por otro lado aplicamos la fórmula de derivación, y luego igualamos:

$$f(t) = t \Rightarrow f'(t) = 1 \Rightarrow f'(2) = 1.$$

$$D_f^{(1)}(2) = a_1 f(0) + a_2 f(2) + a_3 f(4) = a_1 \cdot 0 + a_2 \cdot 2 + a_3 \cdot 4 = 2a_2 + 4a_3.$$

Por lo tanto, nuestra ecuación quedaría

$$1 = 2a_2 + 4a_3.$$

- Veamos la exactitud para  $f(t) = t^2$ . Análogamente:

$$f(t) = t^2 \Rightarrow f'(t) = 2t \Rightarrow f'(2) = 4.$$

$$D_f^{(1)}(2) = a_1 f(0) + a_2 f(2) + a_3 f(4) = a_1 \cdot 0 + a_2 \cdot 4 + a_3 \cdot 8 = 4a_2 + 8a_3.$$

Y finalmente, nuestra última ecuación sería

$$4 = 4a_2 + 16a_3.$$

Resolviendo el sistema de 3 ecuaciones con 3 incógnitas, obtenemos que

$$a_0 = \frac{-1}{4}, \quad a_1 = 0 \quad \text{y} \quad a_2 = \frac{1}{4}. \quad \square$$

### C.3. Derivación numérica utilizando fórmulas de Taylor.

**Ejemplo C.3.** La distancia recorrida por un vehículo se ha ajustado a la siguiente tabla. Use los datos para dibujar la gráfica de esta función y estime la pendiente de la recta tangente en los siguientes puntos:

- a)  $t = 0,5$ , b)  $t = 2,50$ , c)  $t = 1,50$ , d) comparación.

t(horas)	d(Km)
0	0
0.5	38
1	78
1.5	105
2	147
2.5	191

### Solución del ejercicio utilizando Matlab.

- Para ingresar los datos los ponemos en la ventana de *Command Window* de la siguiente manera:

```
t = [0 0.5 1 1.50 2 2.50]';
d = [0 38 78 105 147 191]';
```

Lo ingresamos como matriz transpuesta para que el programa lo utilice como vector.

#### a) Diferencia progresiva en el punto 0.5

Código Matlab
<pre>function[derivacionumerica] = dpro(t, d); plot(t, d) num=2; num1=num+1; %diferencia progresiva     dp = (d(num1) - d(num)) / (t(num1) - t(num)); titulos = ['DERIVADA PROGRESIVA'] derivacionumerica = [ dp ];</pre>

Command Window
<pre>&gt;&gt; dpro(t,d)  titulos = DERIVADA PROGRESIVA  ans =     80</pre>

**b) Diferencia regresiva en el punto 2.50**

Código Matlab
<pre>function [derivacionumerica] = dregre(t, d) plot(t, d) num = 6; num2 = num - 1; %diferencia regresiva     dr = (d(num) - d(num2)) / (t(num) - t(num2)); titulos = ['DERIVADA REGRESIVA'] derivacionumerica=[ dr ];</pre>

Command Window
<pre>&gt;&gt; dregre(t,d) titulos =     DERIVADA REGRESIVA ans =     88</pre>

**c) Diferencia central en el punto 1.50**

Código Matlab
<pre>function [derivacionumerica] = dcen(t, d) plot(t, d) num= 4; num1 = num +1; num2 = num - 1; %diferencia regresiva     dc = (d(num1) - d(num2)) / (t(num1) - t(num2)); titulos = ['DERIVADA CENTRAL'] derivacionumerica = [ dc ];</pre>

Command Window
<pre>&gt;&gt; dcen(t,d) titulos =     DERIVADA CENTRAL ans =     69</pre>

**d) Comparación de los 3 métodos.**

```

Código Matlab
function [derivacionumerica] = diferencianum(t, h)
plot(t, h)
n = length(t);
dp = zeros(n, 1);
dr = zeros(n, 1);
dc = zeros(n, 1);
%diferencia progresiva
for i=1: (n-1)
    dp(i) = (h(i+1) - h(i)) / (t(i+1) - t(i));
end

%diferencia regresiva
for i=2: n
    dr(i) = (h(i) - h(i-1)) / (t(i) - t(i-1));
end

%diferencia central
for i=2: (n-1)
    dc(i) = (h(i+1) - h(i-1)) / (t(i+1) - t(i-1));
end
titulos = ['Tiempo Horas Dprogresiva Dcentral Dregresiva']
derivacionumerica = [t h dp dc dr];

```

Command Window					
>>diferencianum(t, d)					
titulos =					
	Tiempo	Horas	Dprogresiva	Dcentral	Dregresiva
ans =					
	0	0	76.0000	0	0
	0.5000	38.0000	80.0000	78.0000	76.0000
	1.0000	78.0000	54.0000	67.0000	80.0000
	1.5000	105.0000	84.0000	69.0000	54.0000
	2.0000	147.0000	88.0000	86.0000	84.0000
	2.5000	191.0000	0	0	88.0000

**Nota C.1.** *Muy importante destacar diferencianum(t, h), la última función que acabamos de ver en Matlab, que dados dos vectores, t de nodos y h de imagenes o datos, nos calcula una aproximación de la primera derivada. Se anima al lector a implementar las fórmulas de las páginas 62 y 64.*

**Ejemplo C.4.** *Considere la siguiente tabulación de la función  $f(x) = \tan\left(\frac{x}{3}\right)$ , donde los valores están redondeados con cuatro cifras decimales.*

$x$	$f(x) = \tan\left(\frac{x}{3}\right)$
1	0.2463
1,10	0.3849
1,20	0.4228
1.30	0.4627
1.340	0.5038

1. Use la fórmula de diferencias centradas de segundo orden (3.16) con  $h = 0,10$  para encontrar la aproximación de  $f''(1,20)$ .
2. Use la fórmula de diferencias centradas de segundo orden (3.16) con  $h = 0,05$  para encontrar la aproximación de  $f''(1,20)$ .
3. Use la fórmula de diferencias centradas expandidas de cuarto orden de la Tabla de la página 64 con  $h = 0,10$  para encontrar la aproximación de  $f''(1,20)$ .
4. ¿Qué respuesta de los apartados a, b ó c, es más precisa? ¿Porqué?

**Solución:**

$$\begin{aligned}
 a) \quad f''(x_i) &\approx \frac{f(x_i + 0,1) - 2f(x_i) + f(x_i - 0,1)}{h^2}, \\
 f''(1,20) &\approx \frac{f(1,2 + 0,1) - 2f(1,2) + f(1,2 - 0,1)}{0,1^2} \\
 &= 0,110852841757425.
 \end{aligned}$$

$$\begin{aligned}
 b) \quad f''(x_i) &\approx \frac{f(x_i + 0,05) - 2f(x_i) + f(x_i - 0,05)}{0,05^2}, \\
 f''(1,20) &\approx \frac{f(1,2 + 0,05) - 2f(1,2) + f(1,2 - 0,05)}{0,5^2} \\
 &= 0,110774734224250.
 \end{aligned}$$

$$\begin{aligned}
 c) \quad f''(x_i) &\approx \frac{-f(x_i + 2) + 16f(x_i + 1) - 30f(x_i)}{12h^2} \\
 &\quad + 16 \frac{f(x_i - 1) - f(x_i - 2)}{12h^2} \\
 f''(1,20) &\approx \frac{-0,5038 + 16(0,4627) - 30(0,4228)}{12(0,1)^2} \\
 &\quad + 16 \frac{(0,3840) - 0,3463}{12(0,1)^2} \\
 &= \frac{0,0131}{0,12} \\
 &= 0,10916667
 \end{aligned}$$

d) La respuesta es que la aproximación del apartado c) es la más precisa, porque utiliza más términos de la serie de Taylor y tiene un orden 4.

**Resolución en Matlab del ejercicio propuesto.**

```
% Considere la siguiente tabulación de la función  $f(x) = \tan(x/3)$  donde
% los valores están redondeados a cuatro cifras
valoresx = [1 1.1 1.2 1.3 1.4];
valoresf = [0 0.3463 0.3840 0.4627 0.5038];

h = 0.1;
h1 = 0.05;
num = 3;
plot(valoresx, valoresf, 'r', valoresx(num), valoresf(num), 'x');
num1=num+1;
num2=num-1;
num1 = num1 + 1;
num2 = num2 - 1;

disp('Derivada central orden 2, con h=0,10');
derivada = (valoresf(num1) - 2*valoresf(num) + valoresf(num2))/h^2;
disp(derivada);
disp('Derivada central orden 2, con h=0,05');
derivada1 = (valoresf(num1) - 2*valoresf(num) + valoresf(num2))/h1^2;
disp(derivada1);
disp('Derivada central orden 2, extendida con h=0,10');
derivada2 = (-valoresf(num1) + 16*valoresf(num1) - 30*valoresf(num)...
+ 16*valoresf(num2) - valoresf(num2))/(12*h^2);
disp(derivada2);
```

La respuesta de Matlab es:

Command Window
>>Ejemplo3
Derivada central orden 2, con h=0,10 0.110852841757425
Derivada central orden 2, con h=0.05 0.110774734224250
Derivada central orden 2, extendida con h = 0,10 0.10916667

**Ejemplo C.5.** Calcular la derivada de  $f(x) = \sin(x)$  en  $x = \frac{\pi}{4}$  y con  $h = 0,01$ .

a) Diferencias hacia adelante, desde primer orden a cuarto orden

### Código Matlab

```
%Ejemplo de propio basado en el ejemplo de diferencias centrales
%Derivadas numéricas diferencias hacia adelante
%subs -¿sustituye los valores de new en la variable por defecto de la función
%vpa -¿variable-presición arithmetic, muestra todos los decimales o un
%numero de decimales que se le indique, ejemplo -¿(vpa(valor, decimal)).

function y= DiferenciasAdelanteOrdenN
f=input('Ingrese la función f= ');
x0 = input('Ingrese el punto donde desea evaluar la derivada: ');
h=0.01;
f4=subs(f, x0+4*h);
f3=subs(f, x0+3*h);
f2=subs(f, x0+2*h);
f1=subs(f, x0+h);
f0=subs(f, x0);
primeraderivada=(f1-f0)/h;
segundaderivada=(f2-2*f1+f0)/h^2;
terceraderivada=(f3-3*f2+3*f1-f0)/h^3;
cuartaderivada=(f4-4*f3+6*f2-4*f1+f0)/h^4;
disp('Primera derivada hacia adelante')
disp(vpa(primeraderivada))
disp('Segunda derivada hacia adelante')
disp(vpa(segundaderivada))
disp('Tercera derivada hacia adelante')
disp(vpa(terceraderivada))
disp('Cuarta derivada hacia adelante')
disp(vpa(cuartaderivada))
```

### Command Window

```
>>DiferenciasAdelanteOrdenN
Ingrese la funcion f= 'sin(x)'
Ingrese el punto donde desea evaluar la derivada: pi/4
Primera derivada hacia adelante
0.70355949168920325471852532002289
Segunda derivada hacia adelante
-0.71413642493659406062295344563857
Tercera derivada hacia adelante
-0.69641232399643703404697959614188
Cuarta derivada hacia adelante
0.7210945411902455828229525237988
```

### b) Diferencias hacia atrás, desde primer a cuarto orden

Se utiliza la función  $f(x) = \sin(x)$  en  $f(x) = \frac{\pi}{4}$  y con  $h = 0,01$

### Código en Matlab

```
%Ejemplo propio basado en el ejemplo de diferencias centrales
%Derivadas numéricas diferencias hacia atras
%subs -¿sustituye los valores de new en la variable con defectos de la función, luego
%vpa -¿variable-precisión arithmetic, muestra todos los decimales o un
%número de decimales que se le indique, ejemplo -¿(vpa(valor, decimales)).

function y = DiferenciasAtrasOrdenN
f=input('Ingrese la function f= ');
x0 = input('Ingrese el punto donde desea evaluar la derivada: ');
h=0.01;
f4=subs(f, x0-4*h);
f3=subs(f, x0-3*h);
f2=subs(f, x0-2*h);
f1=subs(f, x0-h);
f0=subs(f, x0);
primeraderivada=(f0-f1)/h;
segundaderivada=(f0-2*f1+f2)/h^2;
terceraderivada=(f0-3*f1+3*f2-f3)/h^3;
cuartaderivada=(f0-4*f1+6*f2-4*f3+f4)/h^4
disp('Primera derivada hacia atrás')
disp(vpa(primeraderivada))
disp('Segunda derivada hacia atrás')
disp(vpa(segundaderivada))
disp('Tercera derivada hacia atrás')
disp(vpa(Terceraderivada))
disp('Cuarta derivada hacia atrás')
disp(vpa(cuartaderivada))
```

### Command Window

```
DiferenciasAtrasOrdenN
Ingrese la funcion f= 'sin(x)'
Ingrese el punto donde desea evaluar la derivada: pi/4
Primera derivada hacia atrás
0.71063050057570439140748026419013
Segunda derivada hacia atrás
-0.69999464286271815126476403237083
Tercera derivada hacia adelante
-0.71762446674889305365105108958354
Cuarta derivada hacia adelante
0.69281262689271388621847191365536
```

**Ejemplo C.6.** *Vamos a elegir una sucesión de puntos  $h_n$  de forma que  $h \rightarrow 0$  y analizar que ocurre con*

$$D_n = \frac{f(a + h_n) - f(a)}{h_n}$$

Tomar  $f(x) = e^x$ ,  $a = 1$  y utilizamos  $h_n = 10^{-n}$  para  $n$  desde 1 hasta 18.

Generar una tabla que muestre los valores de  $h$  y los valores de  $D_n$ . ¿Qué observa si  $h$  es muy pequeño?

Código en Matlab
<pre>n=1:18; h=10.^(-n); a=1; x=a+h; valorfn=exp(x); numerador=valorfn-exp(a); DN=numerador./h; format long disp(['Incremento Numerador Aproximación']) disp(['h' numerador' DN'])</pre>

La respuesta de Matlab es:

Command Window		
>>Progresiva 1		
Incremento	Numerador	Aproximación
0.1000000000000000	0.285884195487388	2.858841954873879
0.0100000000000000	0.027319186557871	2.731918655787080
0.0010000000000000	0.002719641422533	2.719641422532781
0.0001000000000000	0.000271841774708	2.718417747078483
0.0000100000000000	0.000027182954199	2.718295419912309
0.0000010000000000	0.000002718283187	2.718283186986525
0.0000001000000000	0.000000271828196	2.718281963964842
0.0000000100000000	0.000000027182818	2.718281777447373
0.0000000010000000	0.000000002718282	2.718281599811689
0.0000000001000000	0.000000000271828	2.718278935276430
0.0000000000100000	0.000000000027183	2.718270053492233
0.0000000000010000	0.000000000002718	2.718270053492233
0.0000000000001000	0.000000000000271	2.713385072183883
0.0000000000000100	0.000000000000027	2.664535259100376
0.0000000000000001	0.000000000000003	2.664535259100376
0.0000000000000000	0	0
0.0000000000000000	0	0
0.0000000000000000	0	0

Como observa en la tabla, mientras más pequeño es  $h$ , el valor de la aproximación devuelto es más pequeño, hasta que en cierta iteración llega a ser cero.

Calculando con diferencia hacia atrás, nos muestra los siguientes valores:

```

Código en Matlab
%Diferencia hacia atrás de f(x)=exp^x en n=18 iteraciones,
%siendo h=10^-n evaluando en 1
n=1:18;
h=10.^(-n);
a=1;
x=a-h;
valorfn=exp(x);
numerador=exp(a)-valorfn;
Dn=numerador./h;
format long
disp(['      Incremento      Numerador      Aproxima-
ción'])
disp(['      Numerador'      Dn'])

```

Command Window		
>>Regresiva1		
Incremento	Numerador	Aproximación
0.1000000000000000	0.258678717302096	2.586787173020957
0.0100000000000000	0.027047356109783	2.704735610978304
0.0010000000000000	0.002716923140479	2.716923140478667
0.0001000000000000	0.000271814591890	2.718145918900738
0.0000100000000000	0.000027182682371	2.718268237122957
0.0000010000000000	0.000002718280470	2.718280469604650
0.0000001000000000	0.000000271828169	2.718281693070423
0.0000000100000000	0.000000027182819	2.718281866265215
0.0000000010000000	0.000000002718282	2.718282043900899
0.0000000001000000	0.000000000271828	2.718283376168529
0.0000000000100000	0.000000000027183	2.718314462413218
0.0000000000010000	0.000000000002719	2.718714142702083
0.0000000000001000	0.000000000000272	2.722266856380884
0.0000000000000100	0.000000000000028	2.753353101070388
0.0000000000000001	0.000000000000003	3.108624468950438
0.0000000000000000	0.000000000000000	4.440892098500626
0.0000000000000000	0	0
0.0000000000000000	0	0

Calculando con diferencia central nos muestra valores más exactos, ya que toma valores de dos puntos para evaluar la función

```

Código en Matlab
%Diferencia central de f(x)=exp^x en n=18 iteraciones,
%siendo h=10^-n evaluando en 1
n=1:18;
h=10.^(-n);
a=1;
x=a-h;
x1=a+h;
valorfn1=exp(x1);
valorfn2=exp(x);
numerador=valorfn1-valorfn2;
Dn=numerador./(2*h);
format long
disp(['      Incremento      Numerador      Aproximación'])
disp(['      Numerador      Dn'])

```

Command Window		
>>Centrall		
Incremento	Numerador	Aproximación
0.1000000000000000	0.544562912789484	2.722814563947418
0.0100000000000000	0.054266542667654	2.718327133382692
0.0010000000000000	0.005436564563011	2.718282281505724
0.0001000000000000	0.000543656366598	2.718281832989611
0.0000100000000000	0.000054365636570	2.718281828517633
0.0000010000000000	0.000005436563657	2.718281828295588
0.0000001000000000	0.000000543656366	2.718281828517632
0.0000000100000000	0.000000054366366	2.718281821856294
0.0000000010000000	0.000000005436564	2.718281821856294
0.0000000001000000	0.000000000543656	2.718281155722480
0.0000000000100000	0.000000000054366	2.718292257952725
0.0000000000001000	0.000000000005437	2.718492098097158
0.0000000000000100	0.000000000000544	2.717825964282383
0.0000000000000010	0.000000000000054	2.708944180085382
0.0000000000000001	0.000000000000006	2.886579864025407
0.0000000000000000	0.000000000000000	2.220446049250313
0.0000000000000000	0	0
0.0000000000000000	0	0

Comparando las 3 tablas, vemos la diferencia que hay con cada una de las 3 derivaciones. Con la diferencia hacia atrás existe la mitad de error de truncamiento que aplicando diferencia hacia adelante o hacia atrás.

Claramente el valor que entrega la diferencia central se ubica en la mitad de los valores que están entre diferencia hacia adelante y diferencia hacia atrás

**Ejemplo C.7. (a)** *La unidad de destello (flash) de una cámara opera por el almacenamiento de carga de un capacitor y su liberación repentina al disparar su unidad. Los datos de la tabla describen la carga  $Q$  que queda en el capacitor (medida en microcoulombios) en el tiempo  $t$  (medido en segundos). use los datos para dibujar la gráfica de esta función y estime la pendiente de la recta tangente en el punto donde  $t=0.04$ .*

T	Q
0.0	100.0
0.02	81.87
0.04	67.03
0.06	54.88
0.08	44.93
0.10	36.76

**(b)** *Un modelo exponencial para la carga es  $Q = e^{4,6053-10,0055x}$ . La derivada  $Q'(t)$  representa la corriente eléctrica que fluye del capacitor hacía el bulbo de la lámpara del destello. Cálcula la corriente cuando  $t = 0,04$ sg. Comparar el resultado con el obtenido en el apartado (a) y rellena la tabla para ver cuál es el porcentaje de error de cada caso.*

**Resultados:**

## Código en Matlab

```
t=[0 0.03 0.04 0.06 0.08 0.1];
q=[100 81.87 67.03 54.88 44.93 36.76];
plot(t, q);
%Posición de instante a estudiar en el vector z
num = 3;
num1 = num+1;
num2 = num-1;
%Diferencia progresiva
disp('Diferencia Progresiva');
difProgresiva = (q(num1)-q(num))/(t(num1)-t(num));
disp(vpa(difProgresiva));
disp('Diferencia Regresiva');
difRegresiva = (q(num)-q(num2)) / (t(num)-t(num2));
disp(vpa(difRegresiva));
disp('Diferencia Central');
difCentral = (q(num1)-q(num2)) / (t(num1)-t(num2));
disp(vpa(difCentral));
%Cálculo de Q0(t)
syms x
derivada=subs(diff(exp(4.6053-10.0055*x)), t(num));
disp('Derivada');
disp(vpa(derivada));

%Comparación
diferencias = [difProgresiva difRegresiva difCentral]

format long
porcentaje = vpa(abs((diferencias - derivada)/derivada)*100);
disp('||||||||||||');
disp('Comparacion');
titulos = [('Progresiva Regresiva')];
disp(titulos);
disp(diferencias : porcentaje)
```

```

Command Window
>>Derivada
Diferencia Progresiva
-607.5

Diferencia Regresiva
-742.0

Diferencia Central
-874.75

Derivada
-672.61293013183037673149612681251

coeficientes =

    -10.0055    4.6053

-----
Comparación
Progresiva          Regresiva          Central
[          -1215/2,          -742,          -2699/4]
[ 9.6805944719302991349425925235178, 10.316047575025050274687401395144, 0.31772655154737556967240443581353]
>>

```

**Ejemplo C.8.** *Hallar la Diferencia hacia adelante, hacia atrás y central ingresando la función y el punto que se desea evaluar la derivada, desde la primera hasta la cuarta derivada*

$$f = (\sqrt{x} + 2x - 5)^{\ln(x+1)}$$

a) Diferencia hacia adelante

## Código en Matlab

```
%Derivadas numericas - diferencias hacia adelante
%subs ->sustituye los valores de new en la variable
%por defecto de la funcion,
%luego se evalua la funcion
%vpa ->variable-precision arithmetic,
%numero de decimales que se le indique.
function y= DifHacDelOrdenN
f=input('Ingrese la funcion f= ')
x0 = input('Ingrese el punto donde desea evaluar la derivada: ')
h=0.001;
f4=subs(f,x0+4*h);
f3=subs(f,x0+3*h);
f2=subs(f,x0+2*h);
f1=subs(f,x0+h);
f0=subs(f,x0);
primeraderivada=(f1-f0)/h;
segundaderivada=(f2-2*f1+f0)/h^2;
terceraderivada=(f3-3*f2+3*f1-f0)/h^3;
cuartaderivada=(f4-4*f3+6*f2-4*f1+f0)/h^4;
disp('Primera derivada hacia adelante')
disp(vpa(primeraderivada))
disp('Segunda derivada hacia adelante')
disp(vpa(segundaderivada))
disp('Tercera derivada hacia adelante')
disp(vpa(terceraderivada))
disp('Cuarta derivada hacia adelante')
disp(vpa(cuartaderivada))
```

### Command Window

```
>> DifHacDelOrdenN
Ingrese la función f= '(sqrt(x) + 2*x-5)^(ln(x+1))'
Ingrese el punto donde desea evaluar la derivada: 2
Primera derivada hacia adelante
2.265711377470083543762682159744

Segunda derivada hacia adelante
1.41251462099839869311470875596

Tercera derivada hacia adelante
5.63272044980700410049042252

Cuarta derivada hacia adelante
-15.176588567617920350363147
```

### b) Diferencia hacia atrás

#### Código en Matlab

```
%Derivadas numéricas - diferencias hacia atrás
%subs -¿sustituye los valores de new en la variable por defecto de la función,
%luego se evalúa la función
%vpa -¿variable-precision arithmetic, muestra todos los decimales o un
%número de decimales que se le indique, ejemplo -> (vpa(valor, decimales)).
function y = DifHaciaAtrOrdenN
f=input('Ingrese la función f= ');
x0=input('Ingrese el punto donde desea evaluar la derivada: ');
h=0.01;
f4=subs(f,x0-4*h);
f3=subs(f,x0-3*h);
f2=subs(f,x0-2*h);
f1=subs(f,x0-h);
f0=subs(f,x0);
primeraderivada=(f0-f1)/h; segundaderivada=(f0-2*f1+f2)/h^2;
terceraderivada=(f0-3*f1+3*f2-f3)/h^3;
cuartaderivada=(f0-4*f1+6*f2-4*f3+f4)/h^4;
```

```
disp('Primera derivada hacia atrás')
disp(vpa(primeraderivada));
disp('Segunda derivada hacia atrás')
disp(vpa(segundaderivada));
disp('Tercera derivada hacia atrás')
disp(vpa(terceraderivada));
disp('Cuarta derivada hacia atrás')
disp(vpa(cuartaderivada));
```

### Command Window

```
>> DifHacAtrOrdenN
Ingrese la función f='(sqrt(x)+2*x-5)^(ln(x+1)) '
Ingrese el punto donde desea evaluar la derivada: 2
Primera derivada hacia atrás
2.252165522945801123615145728379

Segunda derivada hacia atrás
1.2949804092369113365073112656

Tercera derivada hacia atrás
6.13348526312051565442495172

Cuarta derivada hacia atrás
-17.485019643471229286297570
```

### c) Diferencia Central

#### Código en Matlab

```
%Derivadas numéricas - diferencias centrales
%vpa -¿variable-precision arithmetic, muestra todos los decimales o un
%número de decimales que se le indique, ejemplo -> (vpa(valor, decima-
les)).
function y = DifCentralOrdenN
f=input('Ingrese la función f= ');
x0=input('Ingrese el punto donde desea evaluar la derivada: ');
h=0,01;
f2=subs(f,x0+2*h);
f1=subs(f,x0+h);
f0=subs(f,x0);
f_1=subs(f,x0-h);
f_2=subs(f,x0-2*h);
primeraderivada=(f1-f_1)/(2*h);
segundaderivada=(f1-2*f0+f_1)/h^2;
terceraderivada=(f2-2*f1+2*f_1-f_2)/(2*h^3;
cuartaderivada=(f2-4*f1+6*f0-4*f_1+f_2)/h^4;
```

```
disp('Primera derivada central')
disp(vpa(primeraderivada);
disp('Segunda derivada central')
disp(vpa(segundaderivada);
disp('Tercera derivada central')
disp(vpa(terceraderivada);
disp('Cuarta derivada central')
disp(vpa(cuartaderivada);
```

```
Command Window
>> DifCentralOrdenN
Ingrese la función f='(sqrt(x)+2*x-5)^(ln(x+1)) '
Ingrese el punto donde desea evaluar la derivada: 2
Primera derivada central
2.25884150635851761663523996236

Segunda derivda central
1.35472392167766721889802862

Tercera derivada central
5.87506062086288199115569

Cuarta derivada central
-16.79014371150081850263
```

## C.4. Fórmula de Richardson

**Ejemplo C.9.** *Vamos a utilizar la extrapolación de Richardson para mejorar la precisión al momento de calcular la derivada de  $f(z) = \text{sen}(z)$  en  $z = 0,5$  con un paso  $h = 0,01$  y una tolerancia (o error máximo) de  $10^{-6}$ .*

**Solución del ejercicio en Matlab**

## Código Matlab

```
function richardson
global fun
format long
fun = input('ingrese la funcion','s');
x0 = input('ingrese el valor de x0 ');
h= input('ingrese el valor de h ');
tol= input('ingrese la tolerancia ');
err=1;
j=1;
x=x0+h;
f1 = eval(fun);
x=x0-h;
f2 = eval(fun);
D(1, 1)=(f1-f2)/(2*h);
while(err>tol)
h=h/2;
x=x0+h;
f3=eval(fun);
x=x0-h;
f4 = eval(fun);
D(j+1,1)=(f3-f4)/(2*h);
for k=1: j
D(j+1, k+1)=((4^k) * D(j+1, k)- D(j, k)) / ((4^k)-1);
end
err=abs(D(j+1,j+1) - D(j, j));
j=j+1;
end
[ n, n ] = size (D);
D
fprintf('el valor aproximado calculado es:')
D(n,n)
```

## Command Window

```
>>richardson
Ingrese la funcion
sin(x)
ingrese el valor de x0
0.5
ingrese el valor de h
0.01 ingrese la tolerancia
10^(-6)
```

D =		
0.876120655431924	0	0
0.877216948194290	0.877582379115078	0
0.877491149896850	0.877582550464370	0.877582561887655

el valor aproximado calculado es: 0.877582561887655 >>

**Ejemplo C.10.** Use la Extrapolación de Richardson para aproximar  $f'(0,5)$  siendo  $f(x) = 7 + 2,5x - 3,4x^2 + 4x^3$ .

**Solución con Matlab:** Solamente tenemos que usar la función anterior y meter en la ventana de Comandos de Window:

Command Window
>> ExtraRichard Ingrese la funcion f= 7+2.5*x-3.4*(x^2)+4*(x^3) Ingrese el punto donde desea evaluar= 0.5

**Nota C.2.** Se anima al lector a introducir códigos para la segundas derivas, o primeras derivadas de mayor y a experimentar con diferentes valores y diferentes funciones.

## C.5. Ejercicios propuestos

Con los ejemplos (o ejercicios resueltos) de este apéndice el lector puede tener una idea de los problemas que puede resolver y cómo. Además, como ya hemos comentado, la literatura sobre derivación numérica es amplísima y además existe gran variedad de material en Internet. Por tanto, aquí proponemos unos pocos ejercicios al lector a la vez que le animo a la consulta de otros textos. En las referencias del capítulo de derivación numérica podemos encontrar algunos de estos ejercicios así como otros de elaboración propia.

**Ejercicio C.1.** Hallar los coeficientes  $a_i$  para que las siguientes fórmulas de derivación numérica sean exactas en  $\mathbb{P}_k$ :

$$f' \left( \frac{1}{2} \right) \approx a_1 f(0) + a_2 f \left( \frac{1}{2} \right), \quad k = 1.$$

$$f' (3) \approx a_1 f(0) + a_2 f \left( \frac{1}{2} \right) + a_2 f(2) + a_3 f(3), \quad k = 3.$$

$$f'' (1) \approx a_1 f(0) + a_2 f \left( \frac{1}{2} \right) + a_2 f(1), \quad k = 2.$$

**Ejercicio C.2.** Hallar los coeficientes  $a_i$  para que las siguientes fórmulas de derivación numérica sean interpolatorias:

$$f' \left( \frac{a+b}{2} \right) \approx a_1 f(a) + a_2 f(b),$$

$$f'' \left( \frac{a+b}{2} \right) \approx a_1 f(a) + a_2 f(b).$$

¿Qué particularidad tiene la solución de este último apartado? ¿Cuál es el error?

**Ejercicio C.3.** Se necesita calcular la derivada de una función  $f(x) = 4x \operatorname{sen}(x^2)$  en  $x = 1$  y un tamaño de paso  $h = 0,5$ . Realizar el cálculo con todas las fórmulas de derivación vistas en el desarrollo del capítulo.

**Ejercicio C.4.** Dada la función  $f(x) = 4x^3 + x^2 - 5x + 3$ , estime en el punto  $x = 0,5$  empleando un paso  $h = 0,01$  y el algoritmo de Richardson (use una tolerancia  $10^{-9}$ ). Determine el valor real y calcule el error cometido. ¿Qué puede decir al respecto?

**Ejercicio C.5.** Tomar  $f(x) = \frac{x^3-1}{4x}$  y aproximar  $f'(3)$  mediante la fórmula de derivación numérica

$$D_f(x) \approx \frac{f(x+h) - f(x)}{h_n},$$

siendo  $h_n = 5^{-n}$ . Mostrar una tabla con  $n = 1, 2, \dots, 12$  y comparar el error cometido con cada valor de  $n$ . ¿Qué pasa cuando  $n$  es grande?

**Ejercicio C.6.** Para  $f(t) = \ln(t^2)$  calcular aproximadamente  $f(-1)$  por medio de una fórmula de derivación numérica de tipo interpolatorio con tres puntos equiespaciados. Obtener el error real y comparar con el error teórico.

**Ejercicio C.7.** De una determinada función  $f(t)$  se conocen los siguientes datos:

$t$	0	5	10	15	20
$f(t)$	-1,2	3,2	0,75	1	1,9

(a) Estimar  $f'(5)$  y  $f''(15)$  con una fórmula de orden 2.

(b) ¿Puede estimar  $f'(0)$  con una fórmula de orden 2?, ¿y  $f'(10)$ ?, ¿y  $f''(20)$ ?  
En los casos que pueda, hágalo.

**Ejercicio C.8.** Para un circuito con voltaje impreso  $\varepsilon(t)$  e inductancia  $L$ , la primera ley de Kirchoff dice que

$$\varepsilon = L \frac{\partial i}{\partial t} + Ri,$$

donde  $R$  es la resistencia del circuito e  $i$  es la corriente. Se ha obtenido la siguiente tabla:

$t$	1,00	1,01	1,02	1,03	1,04
$i(t)$	3,10	3,12	3,14	3,18	3,24

donde  $t$  se mide en segundos,  $i$  en amperios,  $L = 0,98$  henreis y  $R = 0,142$  ohms. Aproximar el voltaje  $\varepsilon$  en los tiempos  $t = 1$ ,  $t = 1,01$ ,  $t = 1,02$ ,  $t = 1,03$  y  $t = 1,04$  usando las fórmulas apropiadas de tres puntos.

**Ejercicio C.9.** Dada  $f(t) = te^t$  aplicar extrapolación de Richardson para calcular  $f'(2)$  empezando con  $h = 1$  y una tolerancia de  $10^{-8}$ . Realizar lo mismo con  $g(t) = e^{t^2}$ .

*Métodos Numéricos para el Análisis  
Matemático con Matlab*  
Edición digital 2017-2018.  
[www.utmachala.edu.ec](http://www.utmachala.edu.ec)

# Redes

Redes es la materialización del diálogo académico y propositivo entre investigadores de la UTMACH y de otras universidades iberoamericanas, que busca ofrecer respuestas glocalizadas a los requerimientos sociales y científicos. Los diversos textos de esta colección, tienen un espíritu crítico, constructivo y colaborativo. Ellos plasman alternativas novedosas para resignificar la pertinencia de nuestra investigación. Desde las ciencias experimentales hasta las artes y humanidades, Redes sintetiza policromías conceptuales que nos recuerdan, de forma empeñosa, la complejidad de los objetos construidos y la creatividad de sus autores para tratar temas de acalorada actualidad y de demanda creciente; por ello, cada interrogante y respuesta que se encierra en estas líneas, forman una trama que, sin lugar a dudas, inervará su sistema cognitivo, convirtiéndolo en un nodo de esta urdimbre de saberes.



UNIVERSIDAD TÉCNICA DE MACHALA

Editorial UTMACH

Km. 51/2 Vía Machala Pasaje

[www.investigacion.utmachala.edu.ec](http://www.investigacion.utmachala.edu.ec) / [www.utmachala.edu.ec](http://www.utmachala.edu.ec)

ISBN: 978-9942-24-104-7



9 789942 241047