

UNIVERSIDAD DE CANTABRIA

DEPARTAMENTO DE MATEMÁTICAS,
ESTADÍSTICA Y COMPUTACIÓN



TESIS DOCTORAL

PROGRAMACIÓN GENÉTICA,
ALGORITMOS EVOLUTIVOS Y
APRENDIZAJE INDUCTIVO:

HACIA UNA SOLUCIÓN AL PROBLEMA XVII DE
SMALE EN EL CASO REAL.

CRUZ ENRIQUE BORGES HERNÁNDEZ

SANTANDER A, 10 DE FEBRERO DE 2011



Universidad de Cantabria

Departamento de Matemáticas, Estadística y Computación

Programa de Doctorado: Matemáticas y sus Aplicaciones

**Programación Genética, Algoritmos Evolutivos y
Aprendizaje Inductivo:**

Hacia una solución al problema XVII de Smale en el caso real.

Cruz Enrique Borges Hernández

Dirigida por:

*Jose Luis Montaña Arnaiz y
Luis Miguel Pardo Vasallo*

Agradecimientos

Las únicas personas sin problemas están muertas.

Mi abuelo (1928–)

En primer lugar, desearía agradecerle profundamente a *Luis Miguel Pardo* por todos sus sabios consejos a lo largo de estos años. Sin ellos, no sólo gran parte de esta memoria no hubiera sido posible, sino que ni siquiera hubiera sido empezada. En su desarrollo también han tenido mucho que ver *José Luis Montaña* y *César Alonso*, cuyas ideas han perfilado esta memoria.

Tanto como amigos como compañeros de estudio, quiero mostrar mi gratitud a *Cristóbal* y *Rodrigo Camarero* y a *Jose Carlos Tarrío*. Sin ellos hubiera perdido la cabeza en multitud de ocasiones... o a causa de ellos lo he hecho... no lo tengo nada claro.

Agradezco haber contado con el entusiasmo y la cooperación de todo el personal de la biblioteca de la Facultad de Ciencias de la Universidad de Cantabria, así como de los antiguos administradores de sistemas, ahora colaboradores: *Javier Molleda* y *Borja Mantecón*. No puedo tampoco dejar pasar la oportunidad de agradecer la cercanía de *Luis Dubarbie* y *Alicia Nieto* y los demás compañeros del departamento.

Esta tesis ha sido escrita en su totalidad usando software libre. En particular, quiero dar las gracias a los participantes de los proyectos: GNU, Linux, KDE SC, gnuplot, R y T_EXLive por sus contribuciones desinteresadas.

No puedo terminar estas líneas sin agradecer el apoyo incondicional de mi familia y sobre todo de mi mujer. Sin él posiblemente hubiera abandonado ante las primeras dificultades.

Esta memoria ha sido parcialmente financiada por los proyectos MTM2007-62799, MTM2004-01167 y TIN2007-67466-C02-02, DeustoTech y el programa de becas FPU.

Índice general

Índice general	iii
Resolución de Ecuaciones	vii
El modelo de computación bit	xii
El modelo de computación real	xiii
Nuestras contribuciones al programa de Shub y Smale	xvii
Magnitudes puntuales de un sistema de ecuaciones polinomiales	xviii
Algoritmos de deformación homotópica	xxii
Nuestras contribuciones a los algoritmos evolutivos	xxix
El problema de regresión simbólica	xxxiv
Estructura de datos usada para la población	xxxv
Mecanismos de evolución	xxxvi
La elección del modelo	xxxvii
Optimización de las constantes del modelo modelo	xxxviii
Una solución bioinspirada al problema XVII de Smale	xxxix
I La construcción de un algoritmo numérico de deformación homotópica	1
1 Notaciones generales	3
1.1 Normas	4
1.2 La variedad de incidencia, la variedad singular y el triángulo de proyecciones	7
1.3 Algunas fórmulas integrales	9
1.4 El número de condicionamiento	12
2 Algunos resultados integrales básicos	17

3	Norma media de los ceros de un sistema de ecuaciones	25
3.1	Notaciones preliminares	25
3.2	Demostración del Teorema 0.0.32	30
3.3	Demostración del Teorema 0.0.33	31
3.4	Número de soluciones reales de un sistema de ecuaciones	34
4	Distribución de probabilidad del número de condicionamiento	37
4.1	Demostración del Teorema 0.0.16	37
4.2	Demostración del Corolario 0.0.17	42
4.3	Demostración del Corolario 0.0.18	43
5	El número de componentes conexas	45
5.1	Sistemas de ecuaciones lineales con coeficientes reales	45
5.2	Sistemas de ecuaciones polinomiales con coeficientes reales	50
5.2.1	Demostración del Teorema 0.0.27	54
5.3	Sistemas de ecuaciones polinomiales con coeficientes reales y soluciones complejas	54
5.3.1	Demostración del Teorema 0.0.28	57
6	Caminos en la variedad de incidencia	59
7	Trabajos Futuros	63
II	Algoritmos Bioinspirados	65
8	Una <i>vieja</i> estructura de datos para un <i>nuevo</i> problema	67
8.1	La estructura de datos <i>slp</i>	68
8.1.1	Slp's frente a árboles	70
8.1.2	Código efectivo y no efectivo de un <i>slp</i>	72
8.1.3	Cálculo de la aplicación semántica	74
8.2	Programación genética con <i>slp</i> en problemas de regresión	74
8.2.1	La población inicial	75
8.2.2	El operador de calificación	75
8.2.3	Operadores genéticos	76
8.3	Resultados experimentales	80
8.3.1	Operador de cruce para la estructura de datos <i>slp</i>	81
8.3.2	Árboles frente a <i>slp's</i>	83

9	La dimensión de Vapnik-Chervonenkis de un slp	87
9.1	Estimación de la VCD de familias de slp 's	88
9.1.1	Demostración del Teorema 9.0.1	91
9.2	Calificación basada en la VCD de un slp	94
9.3	Criterios para la selección del modelo	96
9.4	Resultados experimentales	97
10	Co-evolución	103
10.1	Programación Genética con slp 's paramétricos	104
10.2	Algoritmos de co-evolución cooperativa	106
10.3	Resultados experimentales	107
10.3.1	Calificación de constantes	107
10.3.2	Algoritmos de co-evolución	109
11	Trabajos Futuros	113
12	Un algoritmo bioinspirado para el problema XVII de Smale	117
12.1	α^* de un sistema de ecuaciones	118
12.1.1	Demostración del Teorema 0.0.47	119
12.2	Algoritmos de búsqueda exhaustiva	120
12.3	Algoritmos evolutivos	122
12.4	Resultados experimentales	124
12.4.1	Protocolo de experimentación	125
12.4.2	Resultados experimentales	127
A	Las funciones Gamma y Beta	131
A.1	Historia de las funciones Γ y β	131
A.2	Propiedades de las funciones Γ y β	134
B	Protocolo de experimentación	137
B.1	Selección del concepto a aprender	137
B.1.1	Polinomios de grado acotado	138
B.1.2	Slp 's de longitud acotada	138
B.2	Generación de una muestra aleatoria simple	139
B.3	Ejecución de las pruebas	139
B.4	Resultados que se muestran	141
	Bibliografía	143

Índice de pseudocódigos	155
Índice de figuras	157
Índice de cuadros	159

Resolución de Ecuaciones

*A man provided with paper, pencil, and rubber, and subject to strict discipline, is in effect a universal machine*¹.

Alan Turing [Tur69]

Los *concursos* de problemas matemáticos siempre han estado en el corazón de la profesión. Desde las colecciones de problemas que se presentan en los libros de texto (y que ya *sufrían* los estudiantes babilónicos) a las diversas *olimpiadas matemáticas* que se organizan todos los años a nivel mundial.

A nivel profesional quizás el primer referente sean las competiciones entre matemáticos del siglo XVI y entre ellas quizás la más sonada sea la disputa entre Tartaglia y del Ferro. En dicha competición se trataba de resolver los problemas propuestos por el contrincante. Estaba en juego tanto la reputación como los fondos universitarios que recibirían. En dicha ocasión los problemas a resolver fueron del tipo: «encuentra las raíces de una ecuación polinomial de tercer grado». Y no es de extrañar dicha elección pues éste es uno de los problemas clásicos de la matemática desde la antigüedad e incluso aún en nuestros días.

En la actualidad seguimos haciendo listas de problemas. Por ejemplo, podemos citar las listas [Hil02], [Sma00] o [CJW06]. Resolver algún problema de dichas listas conlleva un premio, bien en metálico bien con altos honores. . . generalmente ambos. Resulta realmente increíble que el problema anteriormente comentado se encuentre, de una forma u otra, en todas las listas anteriormente citadas. Lógicamente la pregunta ha ido evolucionando. A lo largo de esta introducción veremos cómo y por qué lo ha hecho hasta llegar al enunciado actual: el *Problema XVII de Smale*.

La presente memoria versará sobre dicha versión del problema. Tras la introducción histórica explicaremos el programa de Shub y Smale y el estado del arte sobre dicho programa en el que incluiremos nuestras contribuciones a él. Finalmente introduciremos las técnicas algorítmicas que hemos usado para su resolución y detallaremos el algoritmo, su implementación y sus resultados experimentales. Comenzamos, pues, con la introducción histórica.

¹Un hombre provisto de papel, lápiz y goma y sujeto a una estricta disciplina es, en efecto, una *máquina universal*.

El problema de resolver ecuaciones a lo largo del tiempo

Probablemente los primeros en enfrentarse sistemáticamente al problema de resolver ecuaciones hayan sido los ya mencionados babilónicos. Entre las tablillas de arcilla recuperadas se encuentra listas de problemas en los que se demuestra que ya sabían resolver ecuaciones polinomiales de segundo grado e incluso algunos casos particulares de tercero. Hay que tener en cuenta que lo que entendían como problema solía ser un enunciado geométrico o aritmético y su solución solía ser una lista de casos con los pasos necesarios para encontrar la solución al problema que, obviamente, era numérica².

No es hasta alrededor del siglo IX cuando ³ل-شهورزم escribe uno de los tratados matemáticos más influyentes de la antigüedad: ⁴شْتَبَل-شَبْر و-ل-شَقْبَل. En dicho tratado introduce por primera vez un procedimiento sistemático para la resolución sistemática de ecuaciones cuadráticas y de sistemas de ecuaciones lineales. Nótese que aunque el procedimiento es sistemático, los enunciados de los problemas siguen siendo bastante literarios. La notación algebraica moderna no surgiría hasta mucho después.

El siguiente avance no se produciría hasta el renacimiento. Ya hemos hablado de los retos matemáticos del siglo XVI. En su seno se desarrollaron los algoritmos de resolución de ecuaciones de tercer y cuarto grado debidos a Tartaglia, del Ferro y Ferrari. La publicación de estos métodos por Cardano en su *Artis Magnæ* [Car45] estuvo rodeada de una gran polémica⁵. Para entonces surge una de las primeras grandes incógnitas que ha rondado el problema de resolver ecuaciones polinomiales:

Problema. *¿Pueden resolverse ecuaciones polinomiales de grado mayor que cuatro?*

Entiéndase por resolver, en este contexto, dar una fórmula para las raíces de dichas ecuaciones en la que sólo intervengan las operaciones aritméticas y radicales actuando sobre los coeficientes de los polinomios.

Nótese la sutil diferencia entre esta forma de *resolver* el problema y las anteriores. Ahora se pide una fórmula que relacione los coeficientes de los polinomios con las soluciones mientras que antes simplemente se daba una lista de operaciones que uno debía ejecutar para encontrar la solución buscada. A esta forma de resolver una ecuación se la denomina comúnmente *resolución por radicales*.

La resolución, negativa, a este problema la dio Abel en [Abe26], sin embargo el resultado revolucionario al respecto lo publicó⁶ Galois en [GL46]. En dicho artículo Galois presenta un algoritmo que decide cuando una ecuación puede resolverse mediante radicales, en cuyo caso es capaz de calcular dicha fórmula. Lo novedoso del método es el uso del concepto de *grupo* que más adelante demostraría ser una noción fundamental en múltiples ramas científicas (no sólo en matemáticas). Sin embargo ya el propio Galois se dio cuenta de cual era el punto débil de su algoritmo: la cantidad de operaciones necesarias

²En el sentido de *número* no en el sentido moderno, aunque es posible que ya estuvieran familiarizados con él.

³al-Juarismi.

⁴Compendio sobre el cálculo por completado y balanceo.

⁵Véase [Ste08] para conocer más detalles.

⁶O lo intentó. Véase, otra vez, [Ste08] para una amena narración sobre los trágicos hechos.

para llevarlo a cabo no está al alcance de una persona... ni siquiera usando ordenadores modernos. En la próxima sección hablaremos más sobre este tema.

Ahora bien, sabemos que no hay una fórmula, inmediatamente se presenta el siguiente problema:

Problema. *Aunque no podamos dar una fórmula, ¿tienen raíces todas las ecuaciones?*

Nótese que el problema anterior solicitaba una demostración constructiva y afirmativa a dicho problema.

Esta vez la respuesta, afirmativa, la dio Gauß en [Gau15]. Para ello se apoyó en los entonces exóticos números complejos. De hecho, a lo largo de su vida, presentó hasta cuatro demostraciones distintas para dicho problema. En el contexto actual, la idea detrás de su primera demostración es la siguiente (véase [Sma81] para más detalles):

Sea $f : \mathbb{C} \rightarrow \mathbb{C}$ el polinomio del cual queremos buscar una raíz. Es decir, buscamos un elemento ζ perteneciente a los números complejos tal que $f(\zeta) = 0$. Usando el *Teorema de Morse-Sard* (véase por ejemplo [Hir94]) no es difícil encontrar un elemento $z_0 \in \mathbb{C}$ tal que el segmento $[0, f(z_0)] \subset \mathbb{C}$ no contenga ningún valor crítico de f . Es decir, buscamos un elemento z_0 tal que, para todo z en el conjunto $f^{-1}([0, f(z_0)])$ se cumple que $f'(z)$ es distinto de 0.

Usamos ahora f^{-1} para construir un camino entre los puntos $z_0 \in f^{-1}(f(z_0))$ y $f^{-1}(0) = \zeta$. Su existencia y continuidad está garantizada por el *Teorema del levantamiento de caminos* (véase de nuevo [Hir94]). Hemos demostrado la existencia de una raíz para nuestro polinomio. Sin embargo ni esta, ni ninguna de las demostraciones dadas por Gauß para este problema son constructivas. Es decir, sabemos que existen dichas raíces pero no sabemos calcularlas. Una idea interesante se presenta en el caso anterior: si fuéramos capaces de «seguir» la curva tendríamos un algoritmo que calcularía la raíz. De hecho esta idea ya ha sido desarrollada, como se puede ver en los trabajos de Victor Pan [Pan97] sobre la resolución de ecuaciones polinomiales univariadas. Profundizaremos sobre este tema en las próximas secciones.

Resuelto dicho problema el siguiente paso natural es preguntarse por las soluciones de sistemas de ecuaciones. En el caso de sistemas de ecuaciones lineales, los métodos conocidos por los babilónicos para el caso de dos y tres variables podrían haberse adaptado para el caso general. Pero no es hasta 1809 que Gauß introduce el método que lleva su nombre, en el libro [Gau09], que se resuelve completamente el problema. Sorprendentemente, los matemáticos Chinos ya conocían dicho resultado con casi dos mil años de adelanto. En el libro *九章算術*⁷ escrito alrededor del siglo II antes de la Era Común podemos encontrar un método muy similar al propuesto por Gauß.

En el caso de ecuaciones polinomiales, el procedimiento de *seguir* una curva se puede adaptar para demostrar la existencia de soluciones. De hecho se puede aplicar a cualquier aplicación no lineal diferenciable. Ahora bien, para su correcta aplicación son necesarias herramientas del siglo XX que no estaban disponibles en las fechas que estamos tratando. Es en este contexto en el que Bézout publica en su tesis [Béz79] sus dos principales resultados:

⁷Los nueve capítulos sobre el arte matemático (*Jiǔzhāng Suànshù*).

Teorema (Teorema de Bézout). *Un sistema de ecuaciones polinomiales multivariados con coeficientes complejos tiene tantas soluciones como el producto de los grados de las ecuaciones que forman el sistema.*

Teorema (Identidad de Bézout). *Sea $\{a_1, \dots, a_m\}$ un conjunto de números enteros entonces existen $\{b_1, \dots, b_m\}$ otro conjunto de números enteros tales que se cumple la siguiente igualdad*

$$b_1 a_1 + \dots + b_m a_m = \gcd(a_1, \dots, a_m)$$

Este último resultado es válido para cualquier conjunto de elementos pertenecientes a un *dominio de ideales principales*. En el caso que tomemos como dominio de ideales principales el anillo $\mathbb{C}[X_1, \dots, X_n]$ el enunciado es el famoso *Nullstellensatz*⁸:

Teorema (Nullstellensatz, versión débil (Hilbert [Hil90])). *Sean f_1, \dots, f_m polinomios multivariados con coeficientes complejos. Si denotamos por $V(f_1, \dots, f_m)$ al conjunto de raíces comunes se tiene que: $V(f_1, \dots, f_m)$ es el conjunto vacío si y solo si existen m polinomios g_1, \dots, g_m tales que:*

$$g_1 f_1 + \dots + g_m f_m = 1.$$

La importancia de estos teoremas consiste en que permiten reducir el problema de calcular las soluciones de un sistema de ecuaciones polinomiales multivariado al cálculo de sucesivos determinantes y a la resolución de ecuaciones polinomiales univariadas.

El proceso se basa en usar la *resultante* del sistema de ecuaciones para ir *eliminando* una por una todas las variables. Al final solo queda un polinomio univariado que podemos resolver y posteriormente propagar dichas soluciones. A la rama de las matemáticas encargada de estudiar dichas técnicas se la denomina comúnmente *Teoría de la Eliminación* (véase [vzGG03] o [CLO97] para una introducción a estos temas).

Nótese que hemos transformado el problema de calcular las raíces de un sistema de ecuaciones en un problema de calcular determinantes y resolver ecuaciones polinomiales univariadas, tareas ambas que en este momento ya se sabían resolver. El problema de dichos algoritmos volvía a residir en el número de operaciones necesario para llevarlo a cabo. Insistiremos en este problema en las siguientes secciones pues resultará ser clave.

Resueltos dichos problemas, el siguiente paso era intentar encontrar soluciones para las llamadas ecuaciones diofánticas. Es decir, encontrar soluciones en los números enteros para sistemas de ecuaciones con coeficientes en los números enteros. Este era el problema X de los 23 que Hilbert enunció en [Hil02] como los mayores retos a los que debían enfrentarse los matemáticos durante el siglo XX. De hecho lo fueron, pues dicha lista influyó muy significativamente en la investigación matemática durante la primera mitad del siglo XX.

La lista fue enunciada en una comunicación durante el *Congreso Internacional de Matemáticos* de 1900 celebrado en París. Al principio no tuvo mucha repercusión pero con el tiempo marcó los esfuerzos matemáticos durante la primera mitad del siglo XX. El problema que nos interesa es el llamado *Problema X de Hilbert*:

⁸Teorema de los ceros de Hilbert.

Problema. X - Entscheidung der Lösbarkeit einer Diophantischen Gleichung. *Eine Diophantisch e Gleichung mit irgend welchen Unbekannten und mit ganzen rationalen Zahlen-coefficienten sei vorgelegt: man soll ein Verfahren angeben, nach welchem sich mittelst einer endlichen Anzahl von Operationen entscheiden. läßt, ob die Gleichung in ganzen rationalen Zahlen lösbar ist*⁹.

La resolución de este problema solicitaba idear un algoritmo. Una dificultad añadida radicaba en que en esa época la noción de algoritmo aún no estaba bien fundamentada. No es descabellado asegurar que la búsqueda de una solución a este y al posterior *Entscheidungsproblem*¹⁰ (enunciado también por Hilbert en 1928 [Hil30]) fueron determinantes para que, tanto Alonzo Church como Alan Turing presentaran sus modelos de computación: *Cálculo λ* en el primer caso [Chu36] y la *Máquina de Turing* en el segundo [Tur37b]. Ambos casos son una fundamentación de la idea de algoritmo y de computabilidad equivalentes, tal y como enunciarían, de nuevo ambos casi al mismo tiempo, en la posteriormente llamada *Tesis de Church-Turing* (véase [Tur37a] por ejemplo).

Más adelante von Neumann presenta en [vN45] su «arquitectura» que condujo finalmente a la fabricación física de dichas abstracciones matemáticas. Teniendo en cuenta su origen, no es de extrañar que estas primitivas máquinas se usaran principalmente para la resolución de problemas de índole matemático como descifrar los protocolos criptográficos Nazis (*código enigma*) durante la II guerra mundial (véase [RMH80] para más información al respecto). Pronto aparecieron los problemas:

1. Predecir la cantidad de recursos necesarios para resolver un problema.
2. Estudiar la sensibilidad o estabilidad de los resultados frente a las imprecisiones cometidas.

Para estudiar la primera cuestión se fundamentó el *modelo de computación bit*. Básicamente se trata de estudiar el comportamiento de una *Máquina de Turing* que resuelve el problema. Nos interesa saber la relación entre el tamaño, en bits, de la entrada de dicha máquina y el número de pasos que realiza. El estudio de las cotas superiores e inferiores de dicha relación es el campo de estudio de la *Complejidad Computacional* (véase [Pap94] para una introducción al tema).

La segunda cuestión es el campo de estudio del llamado *Análisis Numérico*. En este contexto el modelo bit es un poco irrealista y no produce muy buenos resultados. Es por ello que Blum, Shub y Smale presentaron en [BSS89] (véase también [BCSS98]) el modelo de computación alternativo BSS con el que se pretende fundamentar el análisis numérico de la misma forma que el modelo bit lo hizo con la complejidad computacional.

Dedicaremos las siguientes páginas a analizar sobre ambos modelos el problema de la resolución de sistemas de ecuaciones polinomiales multivariadas.

⁹Determinación de la resolubilidad de una ecuación diofántica:

Dada una ecuación diofántica con cualquier número de incógnitas y con coeficientes enteros dar un procedimiento tal que determine, en un número finito de operaciones, si la ecuación es resoluble mediante números enteros.

¹⁰Problema de decisión

El modelo de computación bit

El modelo bit de computación ha estado ligado históricamente con la corriente de resolución de ecuaciones de forma simbólica. Dicha corriente es la heredera del discurso que hemos realizado hasta el momento. De hecho, en estos momentos el principal problema de esta corriente se puede enunciar de la siguiente forma:

Problema (III de Smale [Sma00], \mathcal{P} vs. \mathcal{NP} problem [CJW06]). *Given $n + 1$ polinomials in n variables with coefficients in \mathbb{Z}_2 . Decide if there is a common zero $\zeta \in \mathbb{Z}_2^n$.*¹¹

La resolución, en cualquiera de los sentidos, da derecho a reclamar un premio de US\$ 1.000.000 concedido por el *Clay Mathematics Institute* que además provocaría un impacto muy importante en la sociedad actual (véase [For09] para una amena exposición sobre los actuales progresos en la resolución del problema y en sus consecuencias). Como vemos se trata básicamente de una instancia muy particular del *Nullstellensatz*.

Por supuesto, en este contexto, resolver un sistema de ecuaciones polinomiales f consiste en dar un procedimiento algorítmico que a partir de una codificación de los polinomios que forman el sistema, devuelva cualquier información que se solicite sobre el conjunto $V(f)$ de los ceros comunes.

Actualmente, la forma típica de resolver el problema Nullstellensatz es mediante *bases de Gröbner* (véase [CLO97, BW93, GPW03] para un discurso pormenorizado sobre el tema). Sin embargo dicho procedimiento presenta un severo inconveniente: su complejidad es doblemente exponencial. Es decir, la relación entre el tamaño de la entrada (en este caso sistemas de polinomios dados por sus coeficientes) y el número de pasos que son necesarios para resolver dicho problema es del orden $\mathcal{O}(2^{2^n})$ (véase [KM96]). Empíricamente se ha observado que cualquier algoritmo con una complejidad peor que un polinomio (además de grado bajo) es altamente probable que tendrá un desempeño muy malo cuando el tamaño del problema crezca. Aún así, este es el procedimiento implementado en software matemático tan usado como *Maple*TM o *Mathematica*TM.

Para intentar rebajar la complejidad de este problema se intenta afrontar este problema desde otra perspectiva. Como hemos visto anteriormente, para decidir si un sistema de ecuaciones f_1, \dots, f_m tiene una solución común basta con encontrar unos polinomios g_1, \dots, g_m tales que:

$$g_1 f_1 + \dots + g_m f_m = 1.$$

Ahora bien, si somos capaces de acotar los grados de los polinomios g_i y el número de bits que contienen sus coeficiente podremos usar los procedimientos del álgebra lineal para resolver esta cuestión.

La literatura al respecto es muy extensa. A primera vista una cota inferior para la complejidad de este problema es la longitud de los polinomios g_i pues como mínimo debemos escribirlos. Las primeras cotas se deben a Hermann en [Her26]. Desgraciadamente las cotas para los grados de dichos polinomios son del orden $\mathcal{O}(d^{2^n})$ donde d es el máximo de los grados de los polinomios f_i . Esta cota implica que no existe ninguna mejora sobre el método basado en *bases de Gröbner* anteriormente citado. Sin embargo las mejores cotas

¹¹Dados $n + 1$ polinomios in n variables con coeficientes en \mathbb{Z}_2 . Decidir si tienen una raíz común en \mathbb{Z}_2^n .

al respecto son debidas a Krick, Pardo y Sombra (véase [KPS01]) donde presentan cotas superiores para los grados de orden $\mathcal{O}(d^n)$ que si presenta una mejora sustancial sobre los métodos anteriores, aunque sigue sin ser eficientes. Pero es más, dicha cota es bastante ajustada. D. Masser y P. Philippon mostraron una sucesión de sistemas para los cuales una cota inferior para el grado de los polinomios g_i era del orden $\Omega(d^n - d)$.

Para conseguir dicha cota Krick, Pardo y Sombra reutilizan una idea introducida por Ostrowski en [Ost54] cuando estudiaba el problema de la evaluación de polinomios. La idea general es reemplazar la codificación de los polinomios como lista de coeficientes por una especie de *programas* informáticos o, más técnicamente, por una *estrategia de evaluación*. En este contexto un polinomio está codificado por un *grafo dirigido acíclico* y un conjunto de operaciones (típicamente las aritméticas, pero puede ser cualquier otro conjunto) formando lo que en la literatura se denomina un Stright-Line Program (*slp*)¹². La literatura al respecto es amplia y muchos autores han intervenido en el desarrollo de esta estructura de datos (véase por ejemplo [vzG88], [Par95], [GHM⁺98] y [Str90]).

Sin embargo los Stright-Line Programs (*slp*) también puede aplicarse como modelo de computación en si mismo añadiendo al conjunto de funciones los test de signo y, según el modelo, la instrucción de repetición. El uso de este modelo de computación estuvo ligado en un principio al cálculo de cotas inferiores para los problemas de eliminación. En [CGH⁺03], Pardo et ál. (véase también [CPHM01]) propone usarlo tanto como modelo de computación como estructura de datos para polinomios y coeficientes en el problema de resolución de sistemas de ecuaciones polinomiales multivariadas obteniendo un algoritmo óptimo (pues se alcanzan las cotas inferiores enunciadas en la literatura) aunque impracticable, al ser de orden $\mathcal{O}(d^n)$. Una implementación de este algoritmo para el programa de cálculo simbólico MagmaTM se puede encontrar en [DL08].

El modelo de computación real

El modelo de computación real es más reciente en su formalismo. Como ya hemos mencionado, fue propuesto por Blum, Shub y Smale en [BSS89]. En dicho modelo tratan de formalizar un campo de las matemáticas que estaba adquiriendo mucha atención pero que carecía de una fundamentación teórica. Hablamos, por supuesto, del Análisis Numérico. En esta disciplina, se entiende que hemos resuelto un sistema de ecuaciones polinomiales cuando se da un procedimiento que genera una sucesión numérica que converja a la solución.

Los primeros resultados de esta modelo son mucho más antiguos que los presentados en la sección anterior, a fin de cuentas el simbolismo es un invento bastante reciente. El primer caso documentado es posiblemente la tablilla YBC 7289 (véase [Mel]) en la cual se muestra el llamado *método babilónico* (o de Herón) para resolver raíces cuadradas. Dicho procedimiento junto con el conocimiento de la fórmula general posiblemente permitía a los matemáticos babilónicos resolver completamente todas las ecuaciones polinomiales de segundo grado.

¹²En algunos contextos también son conocidos como *circuitos aritméticos*.

El método babilónico prácticamente consiste en aplicar el conocido como *método de Newton* a la ecuación $ax^2 - b = 0$. Dicho método fue introducido por Newton en [New11] para calcular las raíces de ecuaciones cúbicas. Posteriormente trató de usarlo para resolver el problema general de resolución de ecuaciones polinomiales, con un éxito relativo. Newton usaba su método desde un punto puramente algebraico (presentaba una sucesión de polinomios, no una sucesión numérica) y posiblemente esa haya sido la razón de su escaso éxito.

Tuvieron que pasar 30 años hasta que Thomas Simpson reescribiera el método en [Sim37] y lo presenta en su formulación actual. De hecho, lo usó con éxito no solo en ecuaciones polinomiales sino también en contextos analíticos. Es más, descubrió su generalización a varias variables y su uso como método de optimización. En su formulación actual dicho método se describe como:

Definición 0.0.1 (Método de Newton). *Sea $f(x)$ una aplicación con derivada continua. El método de Newton consiste en aplicar sucesivamente el siguiente operador a partir de un punto inicial x_0 :*

$$N_f(x) := x - D_x(f)^{-1}f(x),$$

donde $D_x(f)$ denota el Jacobiano de la función f .

Si x_0 es escogido «suficientemente cercano» a ζ , una raíz de dicha aplicación f , entonces el método converge de forma cuadrática. Es decir, el error h en el paso n entre el valor dado por el método x_n y el valor real ζ es de orden $\mathcal{O}\left(\frac{1}{2^{2^n}}\right)$.

El mayor problema consiste en encontrar buenos candidatos iniciales. Los primeros resultados en este sentido son debidos a Kantorovič (véase [Kan48] y [Ded06] para un desarrollo en términos modernos). En dichos trabajos se presentan los primeros resultados sobre las condiciones necesarias y suficientes para asegurar la convergencia. Sin embargo, los resultados distan mucho de ser satisfactorios.

Durante la década de los 60, S. Smale estaba muy interesado en encontrar un mecanismo que le permitiera hallar raíces reales de sistemas de ecuaciones polinomiales con coeficientes reales pues éstos surgían de forma natural en el estudio de los llamados *equilibrios de Nash* (véase [FT91]). Pero no es la única aplicación importante que tiene. La mayor parte de los métodos de optimización «clásicos» requieren resolver, en alguno de sus pasos, sistemas de ecuaciones polinomiales multivariados. En multitud de ocasiones dichos sistemas provienen de modelos físicos que poseen coeficientes reales y cuyas soluciones interesantes son las reales. Un ejemplo clásico y ampliamente estudiado es el de los modelos de redes eléctricas (véase [BB69]). Es por ello que es sumamente interesante contar con un algoritmo que resuelva dichas sistemas de ecuaciones.

Como ya hemos visto, los algoritmos simbólicos son altamente ineficientes y los numéricos, aunque muy eficientes, presentaban muchos problemas. Para tratar de resolver estas cuestiones, Smale introdujo una fundamentación teórica del análisis numérico: El modelo *BBS* de computación. *Grosso modo*, el modelo es similar a una máquina de Turing a la cual se le permite almacenar número reales en vez de bits. Sorprendentemente el modelo es equivalente al propuesto por Pardo et ál. en [CGH⁺03] basado en *slp's* para la resolución simbólica de sistemas de ecuaciones polinomiales multivariadas (de hecho, el método de Newton se puede considerar como un *slp*).

En la serie de artículos sobre el Teorema de Bézout ([SS93a, SS93b, SS93c, SS94, SS96]) Shub y Smale desarrollan el modelo y presentan un algoritmo no uniforme que calcula raíces de sistemas de ecuaciones polinomiales multivariadas con coeficientes complejos. El método que presentaban funcionaba, a grande rasgos, de la siguiente manera:

Nos basamos en la demostración de Gauß del Teorema Fundamental del Álgebra comentada en la sección anterior. Recordemos, habíamos construido unas curvas desde unas raíces conocidas de un sistema de ecuaciones a las raíces desconocidas del sistema de ecuaciones que estamos estudiando. La idea es usar el método de Newton para construir una poligonal muy próxima a dicha curva y cuyos extremos son:

- Una raíz conocida de un sistema de ecuaciones.
- Un punto suficientemente próximo como para que el método de Newton converja a una raíz del sistema de ecuaciones estudiado.

En la próxima sección analizaremos con más profundidad la construcción de dichas poligonales.

Desgraciadamente los resultados de Shub y Smale al respecto son existenciales, no constructivos¹³. Esto se debe a que no conocían un mecanismo para generar los pares (*sistema, raíz*) necesarios para comenzar el método. Es por ello que, cuando se le solicita a Smale que realice una selección de problemas para el siglo XXI, éste lo introduce en [Sma00] como su problema número XVII:

Problema (XVII de Smale). *Can a zero of n complex polynomial equations in n unknowns be found approximately, on the average, in polynomial time with a uniform algorithm?*¹⁴

En la serie de artículos [BP07b, BP08a, BP09] (véase también [BP]), Beltrán y Pardo desarrollan un algoritmo probabilista que es capaz de generar dichos puntos iniciales y por lo tanto dan una respuesta afirmativa al problema. En esta memoria comenzaremos los pasos preliminares para resolver el caso real, encontrar soluciones reales de sistemas de ecuaciones polinomiales con coeficientes reales, siguiendo el programa de Shub y Smale. Además usaremos algunos de los resultados más importantes para construir diversos algoritmos bioinspirados que experimentalmente dan muy buenos resultados resolviendo dicho problema.

En el siguiente capítulo introduciremos en detalle el programa de Shub y Smale (véase también el Capítulo 1 si no está familiarizado con las notaciones y convenciones de dicho programa) así como nuestras principales contribuciones a él. Finalmente daremos los detalles del algoritmo bioinspirado que proponemos para la resolución del problema XVII de Smale y nuestras contribuciones a las técnicas usadas para el desarrollo del algoritmo.

¹³De ahí la denominación de algoritmo no uniforme.

¹⁴¿Es posible aproximar una raíz de un sistema de n incógnitas y n ecuaciones polinomiales con coeficientes complejos en tiempo polinomial en media?

El algoritmo de deformación homotópica de Shub y Smale: nuestras contribuciones

*If you think it's simple, then you have misunderstood the problem*¹⁵.

Bjarne Stroustrup [Str]

En esta capítulo profundizaremos en la construcción de un algoritmos de deformación homotópica para encontrar soluciones aproximadas de sistemas de ecuaciones polinomiales multivariadas. Además presentaremos nuestras principales contribuciones al respecto.

Como ya hemos mencionando anteriormente, los algoritmos de deformación homotópica, en su versión actual, fueron introducidos por M. Shub y S. Smale en su serie de artículos [SS93a, SS93b, SS93c, SS94, SS96] sobre el *Teorema de Bézout*. En ellos introducen las cantidades a calcular y como usarlas para desarrollar un algoritmo de deformación homotópica con el fin de encontrar ceros aproximados de sistemas de ecuaciones polinomiales multivariadas con coeficientes en los números complejos. Es decir, se trata de resolver el siguiente problema:

Problema 0.0.2 (XVII de Smale, caso complejo general). *Sea $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{C}})$ un sistema de ecuaciones polinomiales multivariado y $\zeta \in \mathbb{C}^n$ una raíz de f . Encontrar, en tiempo polinomial, una sucesión $\{x_i\}_{i \in \mathbb{N}} \in \mathbb{C}^n$ tal que converja cuadráticamente a la raíz ζ . Es decir,*

- $\lim_{i \rightarrow \infty} x_i = \zeta$.
- $\|x_i - \zeta\| \leq \frac{1}{2^{2^i}} \|x_0 - \zeta\|$.

Como se ha comentado con anterioridad el objetivo de este manuscrito es desarrollar un algoritmo que resuelva el Problema 0.0.2 en el que los sistemas de ecuaciones presentan coeficientes reales en vez de complejos. Es decir, tratamos de resolver el siguiente problema:

Problema 0.0.3 (XVII de Smale, caso real general). *Sea $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ un sistema de ecuaciones polinomiales multivariado y $\zeta \in \mathbb{R}^n$ una raíz de f . Encontrar, en tiempo polinomial, una sucesión $\{x_i\}_{i \in \mathbb{N}} \in \mathbb{C}^n$ tal que converja cuadráticamente a la raíz ζ . Es decir,*

¹⁵Si piensas que es simple, entonces no has entendido el problema.

-
- $\lim_{i \rightarrow \infty} x_i = \zeta$.
 - $\|x_i - \zeta\| \leq \frac{1}{2^{2^i}} \|x_0 - \zeta\|$.

Por desgracia, gran parte de los resultados del programa comenzado M. Shub y S. Smale y continuado por L.M. Pardo y C. Beltrán no se pueden aplicar directamente al caso real (Problema 0.0.3) por lo que es necesario reiniciarlo. Siguiendo el mismo esquema, el primer paso es definir el operador de Newton en nuestro contexto.

Definición 0.0.4. *Para todo punto $x \in \mathbb{R}^n$ definimos el operador de Newton como:*

$$N_f(x) := x - D_x(f)^\dagger f(x),$$

donde $D_x(f)^\dagger$ denota la pseudo-inversa de Moore-Penrose de la matriz Jacobiana (véase la Definición 1.4.1). En el caso $m = n$, esta definición coincide con el operador de Newton estándar.

Sea ahora un punto proyectivo $x \in \mathbb{P}_n(\mathbb{R})$. Abusando de la notación, definimos el operador de Newton proyectivo como:

$$N_f(x) := \pi(z - D_z(f)^\dagger f(z)),$$

donde $\pi : \mathbb{R}^{n+1} \setminus \{0\} \rightarrow \mathbb{P}_n(\mathbb{R})$ es la proyección canónica, $D_z(f)^\dagger$ es la pseudo-inversa de Moore-Penrose de la matriz Jacobiana y $z \in \mathbb{R}^{n+1} \setminus \{0\}$ es un punto tal que $\pi(z) = x \in \mathbb{P}_n(\mathbb{R})$. Por supuesto, en el caso $m = n$, esta definición coincide con el operador de Newton proyectivo estándar.

La aplicación iterada del operador de Newton sobre un punto x_0 , cuando está definida, genera una sucesión de valores que llamaremos iteración de Newton. Es decir, llamaremos iteración de Newton a la siguiente sucesión:

$$x_n := N_f(x_{n-1}),$$

que, como es usual, también se denota por

$$x_n := N_f^n(x_0).$$

Magnitudes puntuales de un sistema de ecuaciones polinomiales

El programa de M. Shub y S. Smale comienza introduciendo las siguientes cantidades:

Definición 0.0.5 (α, β, γ) . Sean \mathbb{A} y \mathbb{B} dos espacio de Banach entonces para toda aplicación analítica $f : \mathbb{A} \rightarrow \mathbb{B}$ y para todo punto $x \in \mathbb{A}$ definimos:

$$\begin{aligned} \beta(f, x) &:= \|D_x(f)^{-1} f(x)\|, \\ \gamma(f, x) &:= \sup_{k>1} \left\| \frac{D_x(f)^{-1} D_x^k(f)}{k!} \right\|^{\frac{1}{k-1}}, \\ \alpha(f, x) &:= \beta(f, x) \gamma(f, x), \end{aligned}$$

donde $D_x(f)$ denota la diferencial de f en x . En el caso que $D_x(f)$ no sea suprayectiva entonces tomaremos:

$$\beta(f, x) = \gamma(f, x) = \alpha(f, x) = \infty.$$

Para nuestro discurso tomaremos o bien $\mathbb{A} = \mathbb{R}^n$ y $\mathbb{B} = \mathbb{R}^m$ cuando estemos tratando el caso afín o bien $\mathbb{A} = \mathbb{P}_n(\mathbb{R})$ y $\mathbb{B} = \mathbb{P}_m(\mathbb{R})$ cuando estemos tratando el caso proyectivo. Para este último caso, enunciamos a continuación algunos resultados importantes relacionados con estas cantidades:

Lema 0.0.6. *Usando la notación anterior, a partir de la definición de β se tiene:*

$$\beta(f, x_n) = \|x_{n+1} - x_n\|.$$

Definición 0.0.7 (Cero aproximado). *Sea x_0 un punto tal que la iteración de Newton iniciada en x_0 converge cuadráticamente. Es decir:*

$$\|x_{k+1} - x_k\| \leq \frac{1}{2^{2^k - 1}} \|x_1 - x_0\|. \quad (1)$$

Decimos entonces que x_0 es un cero aproximado.

Teorema 0.0.8 (α -Teorema de Smale, [SS93a, SS93b]). *Sea $x_0 \in \mathbb{P}_n(\mathbb{R})$ un punto tal que*

$$\alpha(f, x_0) < \alpha_0,$$

donde $\alpha_0 := \frac{13-3\sqrt{17}}{4}$. Entonces x_0 es un cero aproximado.

Teorema 0.0.9 (γ -Teorema de Smale, [SS93a, SS93b]). *Sea ζ un cero de la aplicación f . Entonces, para todo punto x_0 tal que:*

$$x \in B(\zeta, r) \text{ con } r < \frac{3 - \sqrt{7}}{2\gamma(f, \zeta)},$$

se tiene que la iteración de Newton converge cuadráticamente. Es decir:

$$\|x_{k+1} - \zeta\| \leq \frac{1}{2^{2^k - 1}} \|x_0 - \zeta\|. \quad (2)$$

Al mejor de los radios r lo llamaremos radio de convergencia y lo denotaremos por $R(f, \zeta)$. Es decir:

$$R(f, \zeta) := \sup\{r \in \mathbb{R}^+ : \forall x \in B(\zeta, r) \text{ } x \text{ es un cero aproximado de } f\}.$$

Nótese la sutil diferencia entre las ecuaciones (1) y (2). Mientras la primera nos da un *test de parada* la segunda da a lugar a la definición del *radio de convergencia*.

Usando los resultados anteriores se puede calcular inmediatamente el número de pasos necesarios para aproximar un cero ζ de una aplicación f con precisión ε suponiendo que tenemos un cero aproximado x_0 . Es decir:

Corolario 0.0.10. *Usando la notación anterior. La distancia de la k -ésima iteración de Newton dista de un cero ζ de f menos de ε siempre y cuando k sea mayor que*

$$k \geq \log \log \frac{C}{\gamma(f, x_0)\varepsilon} + 1,$$

donde C es una constante universal.

Nota 0.0.11. *Nótese que, con las notaciones introducidas anteriormente, el Problema 0.0.3 se puede enunciar de la siguiente manera:*

Problema 0.0.12 (XVII de Smale, caso real versión cero aproximado). *Sea $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ un sistema de ecuaciones polinomiales multivariado y $\zeta \in \mathbb{R}^n$ una raíz de f . Encontrar, en tiempo polinomial, $x_0 \in \mathbb{R}^n$ un cero aproximado de ζ .*

El condicionamiento μ_{norm} es la magnitud clave a la hora de obtener estimaciones de estas cantidades pues sus propiedades geométricas facilitan enormemente su estudio.

El condicionamiento lineal κ_2 ha sido estudiado en profundidad (véase [BT97] y sus referencias). Fruto de este estudio se sabe que está muy relacionado con la pérdida de precisión a la hora de resolver sistemas de ecuaciones lineales. Es decir, nos *mide* lo fácil o difícil que es resolver numéricamente un sistemas de ecuaciones lineales. La extensión a los sistemas de ecuaciones polinomiales multivariadas (véase la Subsección 1.4 para más detalles) fue introducido por M. Shub y S. Smale en [SS93a] y generalizada posteriormente al caso subdeterminado por J. Dégot en [Dég01] y por J.P. Dedieu en [Ded96]. El caso disperso ha sido estudiado por M. Rojas y G. Malajovic en los artículos [MR04] y [BCL06]. Presenta características similares a las del condicionamiento lineal y resulta ser de extrema utilidad pues permite estudiar las cantidades anterior gracias al siguiente teorema:

Teorema 0.0.13 (μ -Teorema de Smale, [SS96]). *Usando la notación anterior:*

$$\gamma(f, x) \leq \frac{D^{\frac{3}{2}}}{2} \mu_{norm}^m(f, x),$$

donde $D = \max\{d_1, \dots, d_m\}$.

Del cual se deduce un corolario inmediato.

Corolario 0.0.14. *Sea ζ un cero de la aplicación f . Entonces, para todo punto x_0 tal que:*

$$x \in B(\zeta, r) \text{ con } r < \frac{3 - \sqrt{7}}{D^{\frac{3}{2}} \mu_{norm}^m(f, x)},$$

se tiene que la iteración de Newton converge cuadráticamente. Es decir:

$$\|x_{k+1} - \zeta\| \leq \frac{1}{2^{2^k - 1}} \|x_0 - \zeta\|.$$

La siguiente proposición que se deduce a partir del μ -Teorema siguiendo un argumento similar al dado por en los Teoremas 130 y 137 de [Ded06] y [BCSS98, Lema 12, página 269] (véase también [SS93a]):

Proposición 0.0.15. *Existe una constante universal u_0 (aproximadamente 0.05992) tal que si $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ es un sistema de ecuaciones polinomiales homogéneas multivariadas con coeficientes reales, $\zeta \in V_{\mathbb{P}}(f)$, $x \in \mathbb{P}_n(\mathbb{R})$ y*

$$d_{\mathbb{P}}(x, \zeta) \leq \min \left\{ \frac{2u_0}{d^{3/2} \mu_{norm}^m(f, \zeta)}, \frac{1}{3} \right\}.$$

entonces existe un cero proyectivo $M_f(x) \in V_{\mathbb{P}}(f)$ tal que:

$$d_{\mathbb{P}}(N_f^k(x), M_f(x)) \leq \frac{2^{3/2} 3}{2^{2k-1}} d_{\mathbb{P}}(x, \zeta).$$

En particular,

$$\min \left\{ \frac{2u_0}{d^{3/2} \mu_{norm}^m(f, \zeta)}, \frac{1}{3} \right\} \leq R(f, \zeta),$$

donde $R(f, \zeta)$ es la cantidad definida anteriormente.

Por todo ello parece esencial el conocimiento de la distribución de probabilidad para conocer el comportamiento de este tipo de operador. Dedicaremos el Capítulo 4 al estudio del condicionamiento μ_{norm} en el caso de sistemas de ecuaciones polinomiales multivariadas con coeficientes en el cuerpo de los números reales. Nuestras principales contribuciones al respecto se resumen en los siguientes teoremas:

Teorema 0.0.16 (Borges y Pardo [BP08b]). *El valor esperado para el condicionamiento μ_{norm} está acotado superior e inferiormente por una cantidad polinomial en \mathcal{D} y N . Es decir: Para todo número real positivo ε tal que $\varepsilon > 0$ definimos la función $Vol_{\varepsilon} : \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \rightarrow \mathbb{R} \cup \{+\infty\}$ como: para todo $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma^{\mathbb{R}}$ tomamos*

$$Vol_{\varepsilon}^m(f) := \nu_{n-m}[\zeta \in V_{\mathbb{P}}(f) : \mu_{norm}(f, \zeta) > \varepsilon^{-1}],$$

donde $\mu_{norm}(f, \zeta)$ denota el condicionamiento μ_{norm} de f en $\zeta \in V_{\mathbb{P}}(f)$ y $\nu_{n-m}[\cdot]$ denota la medida de Hausdorff $n - m$ dimensional en $V_{\mathbb{P}}(f)$.

Sea $2 \leq m$, existen entonces dos constantes universales $c, C \in \mathbb{R}$ ($0.245 \leq c \leq 2$ y $5.013 \leq C \leq 6.414$) tales que, para todo número real ε con $\varepsilon > 0$ y $\varepsilon \leq \frac{1}{n-m+2}$, se cumple:

$$\begin{aligned} \frac{4}{\sqrt{\pi}} \left(\frac{\mathcal{D}}{(n+1)m-1} \right)^{1/2} \left(\frac{(n+1)c\varepsilon}{n-m+2} \right)^{n-m+2} &\leq \\ &\leq E_{\Delta} [Vol_{\varepsilon}^m(f)] \leq \\ &\leq 2 \left(\frac{N\mathcal{D}}{\pi} \right)^{1/2} \left(\frac{(n+1)^{3/2} C \varepsilon}{n-m+2} \right)^{n-m+2}. \end{aligned}$$

En el caso cero dimensional (i.e. $m = n$) podemos definir también el condicionamiento μ peor, que denotaremos $\mu_{worst}(f)$, como el máximo valor que alcanza el condicionamiento μ_{norm} en el conjunto de los ceros $V_{\mathbb{P}}(f)$. Es decir:

$$\mu_{worst}(f) := \max_{x \in V_{\mathbb{P}}(f)} \mu_{norm}^m(f, x).$$

A partir del resultado anterior y de la definición del condicionamiento μ peor el siguiente resultado:

Corolario 0.0.17 (Borges y Pardo [BP08b]). *La distribución de probabilidad de condicionamiento μ peor y su valor esperado están acotados superiormente por una cantidad polinomial en \mathcal{D} y N . Es decir: si $n = m$, entonces para todo número real ε con $0 < \varepsilon \leq \frac{1}{2}$ se cumple:*

- $\text{Prob}_\Delta [f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) : \mu_{\text{worst}}(f) \geq \varepsilon^{-1}] \leq \left(\frac{N\mathcal{D}}{\pi}\right)^{1/2} \frac{(n+1)^3(C\varepsilon)^2}{2}$.
- $\text{E}_\Delta [\mu_{\text{worst}}(f)] \leq 2\sqrt{2} \left(\frac{N\mathcal{D}}{\pi}\right)^{1/4} (n+1)^{3/2}C$.

Corolario 0.0.18 (Borges y Pardo [BP08b]). *El valor esperado para el radio de convergencia esta acotado inferiormente por una cantidad inversamente proporcional a un polinomio en \mathcal{D} , D y N . Es decir: Suponemos que $n = m$, el valor esperado para el radio de convergencia es al menos:*

$$\text{E}_\Delta [R_{\text{worst}}] \geq \frac{2u_0\pi^{1/4}}{6u_0\pi^{1/4} + 2\sqrt{2}D^{3/2}(N\mathcal{D})^{1/4}(n+1)^{3/2}C},$$

donde $D := \max\{d_1, \dots, d_m\}$ es el mayor grado del espacio de sistemas, $5.013 \leq C \leq 6.414$ y $u_0 \approx 0.05992$ es una constante universal.

Nota 0.0.19. *Nótese que el anterior corolario nos provee de una cota inferior de la distancia entre la raíces del sistema.*

Algoritmos de deformación homotópica

Hay que tener en cuenta que las magnitudes anteriores son bastante complejas de computar en la práctica, como veremos en el Capítulo 12. Además sólo proveen información puntual, es decir sólo nos dan información acerca de si un punto es, o no, un cero aproximado pero no dice nada acerca de como encontrarlo. En el Capítulo 12 presentamos un algoritmo en el que usando técnicas evolutivas buscamos un cero aproximado usando dichas cantidades similares como test.

Sin embargo, otra forma de encontrar un cero aproximado es mediante una *deformación homotópica*. Dicho método funciona como sigue: Sea $\varphi : [0, 1] \rightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{C}})$ una curva tal que:

- $\varphi(0) = g$ sea un sistema con algunas características *apropiadas* y $\varphi(1) = f$ el sistema del cual queremos encontrar un cero aproximado.
- Para todo $t \in [0, 1]$, $\varphi(t)$ es un sistema de ecuaciones que posee, al menos, una raíz de multiplicidad 1.

A partir de esta curva se tiene definida de forma implícita otra curva en la variedad de incidencia vía el *Teorema de levantamiento de caminos*.

$$\begin{aligned} \varphi' : [0, 1] &\longrightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{C}}) \times \mathbb{P}_n(\mathbb{C}) \\ t &\longmapsto (f_t, \zeta_t) \end{aligned}$$

tal que:

-
- Para todo t en $[0, 1]$, $f_t(\zeta_t) = 0$.
 - Existe $\zeta_0 \in V_{\mathbb{C}}(f_0)$ tal que para todo t en $[0, 1]$, ζ_t no es una raíz múltiple de f_t .

Como consecuencia inmediata del γ -teorema (Teorema 0.0.9) existe un tubo de radio mayor que:

$$R^* = \min_{t \in [0,1]} \{R(f_t, \zeta_t)\},$$

en el cual para todo punto x_t perteneciente a la bola $B(\zeta_t, R^*)$, se tiene que x_t es un cero aproximado de f_t . Es decir, la iteración de Newton convergerá a ζ_t . La idea central de un algoritmo de deformación homotópica se basa en *seguir* la curva anterior. Para ello debemos escoger:

- Un número natural k .
- Una partición de $[0, 1]$, $t_0 = 0 < t_1 < \dots < t_k = 1$ que a su vez genera una sucesión de sistemas $\varphi(t_i) := f_{t_i}$ con $\varphi(0) := g$ y $\varphi(1) := f$.

Recordemos que $g \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{C}})$ es un sistema de ecuaciones del cual conocemos al menos un cero aproximado x_0 y $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{C}})$ es el sistema del cual queremos conocer una (o varias) de sus raíces. La idea es escoger los valores anteriores de forma que la siguiente sucesión:

$$x_n := N_{f_{t_n}}(x_{n-1})$$

esté contenida dentro de dicho tubo. Más concretamente:

$$\forall n, 1 \leq n \leq k : x_{n-1} \in B(\zeta_{t_n}, R^*).$$

Nótese que esto nos permite *seguir* la curva pues vamos *saltando* de un cero aproximado a otro hasta obtener un cero aproximado del sistema deseado. A toda sucesión que cumpla dicha propiedad la denominaremos sucesión de Newton. Véase la Figura 1 para un ejemplo esquemático.

La forma más sencilla de homotopía es la llamada *homotopía lineal*. Ya estaba indicada en el programa de Shub y Smale [SS93a] y consiste en tomar:

- $t_0 = 0 < \dots < t_i = i/k < \dots < t_k = 1$.
- $\varphi(t) = (1 - t)g + tf$.

El problema consiste en estimar el número k tal que lo anterior sea una sucesión de Newton. La primera demostración de que dicho k existía fue dada por Shub y Smale en [SS94]:

Teorema 0.0.20 (Existencia de sucesiones de Newton). *Para todo $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{C}})$ y para todo número real $\varepsilon > 0$ existe un k que depende polinomialmente de n , N , D y del condicionamiento μ_{norm} tal que si tomamos la homotopía lineal con k pasos entonces la sucesión de Newton asociada a dicha partición produce un cero aproximado x_k de un cero ζ de f .*

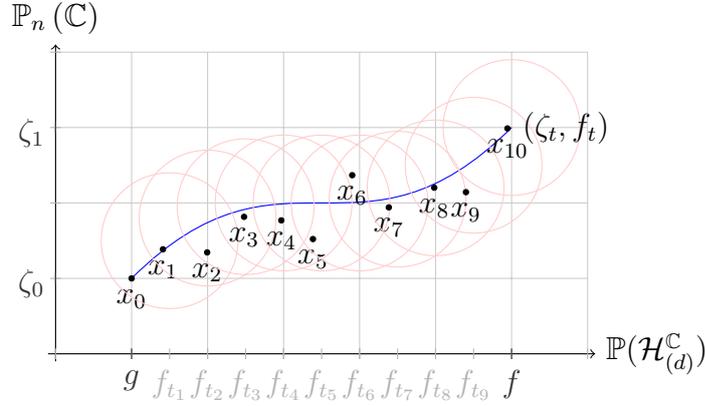


Figura 1: *Sucesión de Newton.*

El número total de pasos por lo tanto esta acotado por:

$$k + 1 + \log \log \frac{C}{\varepsilon \gamma(f, x_k)}.$$

Otras formas de construir la homotopía se han estudiado en [BS09, BP] que mejoran enormemente las cotas, haciendo casi lineal el método.

Nota 0.0.21. *Nótese que, con los resultados introducidas anteriormente, el Problema 0.0.3 se puede enunciar de la siguiente manera:*

Problema 0.0.22 (XVII de Smale, caso real versión homotopía lineal). *Sea $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ un sistema de ecuaciones polinomiales multivariado y $\zeta \in \mathbb{R}^n$ una raíz de f . Encontrar, en tiempo polinomial, un par $(g, x_0) \in V$ tal que la homotopía lineal asociada genere una sucesión de Newton.*

Se nos presentan ahora varios problemas:

1. Cómo seleccionar un buen candidato inicial (g, x_0) .
2. Cómo encontrar ceros afines.

Shub y Smale eran conscientes de estos problemas. Para el primero de dichos problemas ofrecieron la siguiente conjetura en [SS94]:

Conjetura 0.0.23. *El siguiente sistema produce siempre una sucesión de Newton cuando se usa como candidato inicial usando la homotopía lineal.*

$$\begin{aligned} d_1^{1/2} X_0^{d_1-1} X_1 &= 0 \\ &\vdots \\ d_n^{1/2} X_0^{d_n-1} X_n &= 0 \end{aligned}$$

En el caso de sistemas de ecuaciones con coeficientes complejos dicha conjetura sigue siendo un problema abierto a día de hoy. Sin embargo, en el caso de coeficientes reales la conjetura no es cierta como demostraremos más adelante (para más detalles véase el Capítulo 5).

Otra forma de afrontar dicho problema fue el trabajo realizado por C. Beltrán y L.M. Pardo en la serie de artículos [BP07b, BP08a, BP09] que solventa el problema de encontrar un sistema inicial reusando una idea sorprendente y revolucionaria que da lugar a un algoritmo probabilista (para ver un resumen con los últimos avances consultar [Bel]). Esta idea consiste en definir un conjunto cuestor $\mathcal{G} \subseteq \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{C}})$ (para más información acerca de los conjuntos cuestores véase [GH93], [HS80] y [KP94]) que tiene la siguiente propiedad:

Teorema 0.0.24. *Si se escoge al azar un elemento de \mathcal{G} como candidato inicial para la homotopía lineal entonces con alta probabilidad se genera una sucesión de Newton con un número de pasos polinomial en n , N y D .*

Nota 0.0.25. *En particular, la Conjetura 0.0.23 afirma que existe un conjunto cuestor universal de cardinal 1. Es más, en este contexto el Problema 0.0.3 se puede enunciar de la siguiente manera:*

Problema 0.0.26 (XVII de Smale, caso real versión cuestor). *Sea $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ un sistema de ecuaciones polinomiales multivariado y $\zeta \in \mathbb{R}^n$ una raíz de f . Generar, en tiempo polinomial, un conjunto cuestor de puntos iniciales para la homotopía lineal que comprueba la propiedad ser una sucesión de Newton.*

Dedicaremos los Capítulos 5 y 6 a estudiar las componentes conexas del sistema de ecuaciones polinomiales homogéneas multivariadas con coeficientes reales y las posibles formas de definir caminos en ellas pues debido a sus nefastas particularidades no es factible *a priori* que la homotopía lineal funcione. Nuestros principales resultados al respecto son:

Teorema 0.0.27 (Borges y Pardo). *Las componentes conexas de la variedad de incidencia sin el conjunto de soluciones singulares $\Sigma'^{\mathbb{R}}$ quedan caracterizadas de la siguiente forma:*

- Si $m < n$, entonces $\beta_0(V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}) = 1$.
- Si $m = n$ y n es impar entonces $\beta_0(V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}) = 1$.
- Si $m = n$ y n es par entonces $\beta_0(V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}) = 2$.

Es más, en este último caso las componentes conexas son:

$$V^{\mathbb{R}^+} = \{(f, x) \in V^{\mathbb{R}} : \det(\Delta_x f) > 0\},$$

$$V^{\mathbb{R}^-} = \{(f, x) \in V^{\mathbb{R}} : \det(\Delta_x f) < 0\},$$

donde $\Delta_x f$ denota el jacobiano extendido orientado (véase la Definición 5.2.4).

Teorema 0.0.28 (Borges y Pardo). *El número de componentes conexas del espacio de sistema de ecuaciones polinomiales multivariadas sin la variedad discriminante está acotado inferiormente por:*

- $\mathcal{D} \leq \beta_0 \left(\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma \right)$.
- $\mathcal{D} \leq \beta_0 \left(V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma' \right)$.

donde $\mathcal{D} = \prod_{i=1}^n \deg(f_i)$ denota el número de Bézout.

Nota 0.0.29. *A partir del corolario anterior se deduce inmediatamente que la Conjetura 0.0.23 no es cierta en el caso de sistemas de ecuaciones polinomiales multivariadas con coeficientes reales pues existen casos en los que la variedad de incidencia tiene más de una componente conexa.*

Para el segundo problema, la norma media de las soluciones afines de un sistema de ecuaciones polinomiales multivariadas se usó en [BP09] para conseguir transformar una solución proyectiva en una solución afín. El procedimiento se basa en los siguientes teoremas:

Proposición 0.0.30 (Beltrán y Pardo [BP09]). *Sea $f \in \mathcal{H}_{(d)}^{\mathbb{C}}$ un sistema de ecuaciones polinomiales homogéneas multivariadas con $\zeta \in V_{\mathbb{P}}(f)$ un cero proyectivo de f tal que $\zeta_0 \neq 0$. Sea $z := (z_0, \dots, z_n) \in \mathbb{P}_n(\mathbb{C})$ un cero aproximado proyectivo de f con cero asociado ζ . Sea $z^k := N_f^k(z)$ el punto proyectivo obtenido después de k pasos del operador de Newton con $k \in \mathbb{N}$ tal que:*

$$k \geq \log \log(4(1 + \|\varphi_0^{-1}(\zeta)\|^2)),$$

entonces $\varphi_0^{-1}(z^k)$ es un cero aproximado de f con cero asociado $\varphi_0^{-1}(\zeta)$.

Teorema 0.0.31 (Beltrán y Pardo [BP09]). *Para todo número real ε menor que 1 la probabilidad de que la norma de los ceros afines de un sistema $f \in \mathbb{P}(\mathcal{P}_{(d)}^{\mathbb{C}})$ sea mayor que $\frac{\varepsilon}{\mathcal{D}} \sqrt{\pi n}$ es menor que ε .*

Dedicaremos el Capítulo 3 al comportamiento de la norma de las soluciones afines de sistemas de ecuaciones polinomiales multivariadas con coeficientes reales (versión real del Teorema 0.0.30). En particular demostraremos los siguientes resultados:

Teorema 0.0.32 (Borges y Pardo [BP08b]). *El valor esperado para la norma de los ceros de un sistema de ecuaciones polinomiales multivariados con coeficientes reales es proporcional a una función hipergeométrica. En otras palabras, sea r un número real tal que $0 < r < 1$. Denotaremos por $N_{av}^r(f)$ a la esperanza de la norma de los ceros afines en $V_{\mathbb{A}}^{\mathbb{R}}(f)$, Es decir,*

$$N_{av}^r := E_{V_{\mathbb{A}}^{\mathbb{R}}(f)} [\|\cdot\|^r].$$

Entonces, se cumple:

$$E_{\Delta} [N_{av}^r(f)] = \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right) \frac{\nu_{\Delta}[\mathcal{R}]}{\vartheta_N}.$$

donde $\beta(x, y)$ denota la función hipergeométrica de mismo nombre (también llamada integral de Euler de primera especie, véase Anexo A para más detalles) y $\vartheta_k = \nu_k[\mathbb{P}_k(\mathbb{R})]$ denota el volumen del espacio proyectivo real de dimensión k .

En particular, si todos los elementos de la lista de grados $(d) = (d_1, \dots, d_m)$ son números impares, entonces

$$E_{\Delta} [N_{av}^r(f)] = \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right).$$

Teorema 0.0.33 (Borges y Pardo [BP08b]). *El valor esperador para la norma media de la variedad solución de un sistema de ecuaciones polinomiales multivariados con coeficientes reales es proporcional a la raíz cuadrada de \mathcal{D} por el volumen de un espacio proyectivo. Es decir, para todo número real r tal que $0 < r < 1$ se tiene:*

$$E_{\Delta} \left[\int_{x \in V_{\mathbb{A}}(f)} \|x\|^r dV_{\mathbb{A}}(f) \right] = \mathcal{D}^{1/2} \vartheta_{n-m} \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right).$$

donde $\beta(x, y)$ denota la función hipergeométrica de mismo nombre y ϑ_k denota el volumen del espacio proyectivo real de dimensión k .

Corolario 0.0.34 (Borges y Pardo [BP08b]). *Con alta probabilidad todas las raíces de un sistema de ecuaciones polinomiales multivariadas están contenidas en una bola de tamaño polinomial en \mathcal{D} . Es decir, si $n = m$ con $n \geq 3$ y ε es un número real positivo tal que $\varepsilon < 1$, entonces la siguiente desigualdad se cumple:*

$$\text{Prob}_{\Delta} \left[\max_{\zeta \in V_{\mathbb{A}}^{\mathbb{R}}(f)} \|\zeta\| \geq \varepsilon^{-1} \right] \leq \Gamma \left(\frac{1}{4} \right) \mathcal{D}^{1/2} \varepsilon^{1/2},$$

donde $\Gamma(x)$ denota la función hipergeométrica de mismo nombre (también llamada integral de Euler de segunda especie, véase Anexo A para más detalles) y \mathcal{D} denota el número de Bézout, es decir, $\mathcal{D} = \prod_{i=1}^m d_i$.

Otra cantidad sumamente interesante es el volumen medio que tiene la variedad algebraica de los ceros reales de un sistema de ecuaciones polinomiales multivariadas. Esta cantidad tiene interés por si misma, pero en nuestro caso la necesitaremos a la hora de estimar muchas de las cantidades anteriores en el caso real. El caso cero dimensional fue estudiado por M. Shub y S. Smale en [BCSS98] y el siguiente resultado presenta su generalización al caso subdeterminado.

Teorema 0.0.35 (Kostlan, Prior, Bürgisser ([Kos02],[Pri06])). *Sea m, n dos números naturales tales que $m \leq n$. Entonces, el valor esperado para el volumen de la variedad real del sistema es $\mathcal{D}^{1/2} \nu_{n-m}$. Es decir:*

$$E_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} [\nu_{\Delta}[V_{\mathbb{P}}^{\mathbb{R}}(f)]] = \frac{1}{\nu_N} \int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \nu_{\Delta}[V_{\mathbb{P}}^{\mathbb{R}}(f)] d\nu_{\Delta} = \mathcal{D}^{1/2} \nu_{n-m}$$

Dedicaremos la Sección 3.4 a demostrar dicho resultado pues resulta inmediata a partir de las técnicas desarrolladas en el Capítulo 2.

Algoritmos evolutivos para la resolución de problemas de optimización: nuestras contribuciones

*Une théorie, un système, une hypothèse, l'Évolution?... Non point: mais, bien plus que cela, une condition générale à laquelle doivent se plier et satisfaire désormais, pour être pensables et vrais, toutes les théories, toutes les hypothèses, tous les systèmes. Une lumière éclairant tous les faits, une courbure que doivent épouser tous les traits: voilà ce qu'est l'Évolution.*¹⁶

Pierre Teilhard de Chardin [dC56]

Muy relacionados con el problema que hemos estudiado en el capítulo anterior se encuentran los llamados *problemas de optimización*. Se trata ahora de encontrar un valor que maximice o minimice una aplicación suponiendo que cumple algunas restricciones. En general, resolver un sistema de ecuaciones puede verse como una instancia particular de un problema de optimización de la siguiente forma: transformamos

Problema. *Dado f un sistema de ecuaciones, encontrar x tal que $f(x) = 0$,*

en la siguiente instancia de un problema de optimización:

Problema. *Minimizar $\|f\|$ suponiendo que x pertenece a todo el espacio de búsqueda.*

Esta dicotomía se observa claramente en los llamados *Métodos Clásicos* en los cuales transformamos un problema de optimización en uno de resolución de sistemas de ecuaciones. Dentro de esta categoría se engloban métodos con una larga tradición. Los primeros probablemente datan del siglo XVIII como por ejemplo: Descenso de Gradiente, formulado por Gauß o los Multiplicadores de Lagrange. Sin embargo, su auge se produjo durante la II Guerra Mundial cuando George Dantzig introdujo en [Dan63] el concepto de *Programación Matemática* en el contexto de *programar* acciones logísticas o bélicas de forma *óptima*.

En general, todos los métodos enunciados anteriormente necesitan (de hecho se basan en) resolver un sistema de ecuaciones que depende de la aplicación f a optimizar y de las

¹⁶¿Es la evolución una teoría, un sistema o una hipótesis? Es mucho más: es una condición general ante la cual todas las teorías, todas las hipótesis, todos los sistemas deben inclinarse y satisfacer a partir de ahora si pretender ser razonables y verdaderas. La evolución es la luz que ilumina todas las verdades, una curva que todas las líneas deben seguir.

condiciones g que impongamos a las soluciones. En el caso que f y g sean aplicaciones lineales, existen métodos eficientes para resolver el problema, de la misma forma que existen métodos eficientes para resolver sistemas de ecuaciones lineales. Sin embargo, cuando las aplicaciones f y g son polinomiales nos encontramos con la necesidad de resolver sistemas de ecuaciones polinomiales que, como ya hemos visto en la sección anterior, siguen sin saberse resolver eficientemente.

Es por este motivo por el que surgen toda una serie de técnicas algorítmicas, generalmente llamadas *heurísticos* que, si bien no garantizan su complejidad u optimalidad, si que presentan en la práctica resultados suficientemente buenos. Es en este grupo en donde se encuadran los *algoritmos bioinspirados*. Nuestro interés en los problemas de optimización viene motivado por dichos algoritmos bioinspirados pues, como veremos más adelante, son capaces de resolver en la práctica el problema de optimización que se deriva del problema XVII de Smale. Es decir, son capaces de resolver el siguiente problema:

Problema 0.0.36 (XVII de Smale, caso problema optimización). *Sea $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ un sistema de ecuaciones polinomiales multivariado, encontrar un punto tan próximo a una raíz del sistema como se pueda.*

En las siguientes secciones estudiaremos los problemas de optimización en general, nuestras contribuciones al respecto, para finalmente discutir la construcción de varios algoritmo evolutivo que son capaces de resolver el problema XVII de Smale en el caso real.

Algoritmos Bioinspirados

Los algoritmos bioinspirados son heurísticos que resuelven problemas de optimización. Se inspiran en procesos naturales que, a fin de cuentas, no son más que estrategias de optimización que lleva funcionando más de 3.000 millones de años y que consiguen resolver, de alguna forma, los problemas a los que se enfrenta.

En general, un algoritmo bioinspirado se compone de los siguientes elementos:

Población: Conjunto de candidatos a ser solución que pueden, o no, cumplir las restricciones del problema.

Mecanismo de Evolución: Conjunto de procedimientos por los cuales se modifica la población.

Mecanismo de calificación: Conjunto de procedimientos por los cuales se asigna una calificación a cada elemento de la población. Usualmente es la función a optimizar.

Condiciones de Parada: Mecanismo que controla el número de operaciones que puede ejecutar el algoritmo.

El pseudocódigo típico de un algoritmo de este tipo es muy similar al siguiente:

Pseudocódigo 1: *Pseudocódigo de un método evolutivo.*

```
inicializar población
mientras (no condición de parada) hacer
    evolucionar población
    calificar población
    reemplazar población
fin
```

Estos mecanismos no son exclusivos de los algoritmos bioinspirados pues muchos de los métodos considerados clásicos encajan en esta descripción.

De los mecanismos anteriores, usualmente, la condición de parada, la población y el mecanismo de calificación vienen determinados por el problema a resolver, aunque no siempre es así. Sin embargo, los mecanismos de evolución son característicos de cada método. Teniendo en cuenta esto, detallamos los mecanismos de evolución de algunos de los algoritmos bioinspirados (véase [ES03], [Koz92] y [OZ05] para una descripción detallada de cada método):

Algoritmos de Escalada: A partir de una solución (o casi solución), se producen pequeñas modificaciones que sólo se aceptan si mejoran la calificación. En el caso de que permitamos también movimientos que empeoren la calificación (asignando probabilidades de aceptar uno u otro caso) el método se suele denominar *Enfriamiento Simulado*. Véase [LA87] para una introducción a los algoritmos basados en esta heurístico.

El pseudocódigo de este tipo de algoritmos es muy similar al siguiente:

Pseudocódigo 2: *Pseudocódigo de un algoritmo de escalada.*

```
generar población inicial
calificar población inicial
mientras (no condición de finalización) hacer
    desde  $i=1$  hasta tamaño de la población hacer
        generar vecino
        calificar vecino
        si vecino tiene mejor calificación que el original
            añadir a la nueva población
        en otro caso
             $Op :=$  valor aleatorio entre 0 y 1
            si ( $Op < Probabilidad\ de\ Reemplazo$ )
                añadir a la nueva población
    fin
    sustituir la población con la nueva población
fin
```

Algoritmos Evolutivos: Parten de una población completamente aleatoria a la cual se le aplican *Operadores Genéticos*:

Operador de selección: Selecciona los individuos sobre los que se va a operar.

Operador de reproducción: Selecciona los individuos que sobreviven a una operación genética.

Operador de cruce: Crea dos elementos nuevos combinando las características de dos individuos preexistentes.

Operador de mutación: Crea un individuo modificando otro preexistente.

De este modo se produce una evolución, similar a la de las especies, en la que solo los mejores calificados pasan a la siguiente generación.

Dentro de este apartado, cabe reseñar que cuando el objeto de evolución son estructuras similares a *programas informáticos* se suele denominar *programación genética*.

El pseudocódigo este tipo de algoritmos es muy similar al siguiente:

Pseudocódigo 3: *Pseudocódigo de un algoritmo genético.*

```
generar población inicial
calificar población inicial
mientras (no condición de finalización) hacer
  desde  $i=1$  hasta tamaño de la población hacer
     $Op :=$  valor aleatorio entre 0 y 1
    si ( $Op < Probabilidad\ de\ cruce$ ) entonces
      seleccionar dos individuos
      aplicar operador de cruce
    en otro caso
      si ( $Op < Probabilidad\ de\ cruce + Probabilidad\ de\ mutación$ )
        entonces
          seleccionar un individuo
          aplicar operador de mutación
        en otro caso
          seleccionar un individuo
          aplicar reproducción
      calificar a los nuevos individuos
      esoger individuos para la nueva población
      añadir a la nueva población
    fin
  fin
  sustituir la población con la nueva población
fin
```

En la Capítulo 8.1 estudiaremos con mucho más detalle este heurístico.

Enjambres de Partículas: Teniendo en cuenta que cada elemento de la población es una posición en el *espacio de posibles estados*, la idea es que cada partícula o elemento se mueva por ese espacio de forma más o menos aleatoria. Las partículas podrán tener *memoria* y podrán intercambiar *información* con el resto de partículas del enjambre o población.

El concepto de memoria e información suele estar ligado al mecanismo de calificación mientras que el movimiento (que depende de la memoria y la información de las partículas vecinas) lo esta con el mecanismo de evolución. Véase [Pol08] para una introducción a los algoritmos basados en esta heurístico.

El pseudocódigo este tipo de algoritmos es muy similar al siguiente:

Pseudocódigo 4: *Pseudocódigo de un algoritmo de enjambre de partículas.*

```
generar población inicial
calificar población inicial
inicializar posiciones con mejor calificación encontrada
    por cada individuo
inicializar posición con mejor calificación encontrada
    por la población
mientras (no condición de finalización) hacer
    desde  $i=1$  hasta tamaño del población hacer
        mover individuo
        calificar nuevo individuo
        si calificación nuevo individuo mejor que original
        entonces
            añadir a la nueva población
            si calificación nuevo individuo mejor que
            mejor calificación encontrada por el individuo
            entonces
                actualizar mejor calificación individuo
                si calificación nuevo individuo mejor que
                mejor calificación población
                entonces
                    actualizar mejor calificación población
        fin
    sustituir el población con el nuevo población
fin
```

La gran ventaja de todos estos métodos es su *universalidad*. Cualquier problema de optimización que pueda ser finitamente parametrizado es susceptible de ser optimizado mediante estas técnicas. La única condición es que se pueda dar una función de calificación que efectivamente recoja la función que se pretende optimizar. Este hecho los hace muy apropiados para gran número de dominios como se puede ver en [Wik10].

Aunque estemos tratando con técnicas de tipo heurísticas, al menos en el caso de los algoritmos evolutivos y la programación genética, existe cierta fundamentación teórica. Antes de enunciarla necesitamos una definición.

Definición 0.0.37 (Esquema). *Sea $x := (x_1, \dots, x_n)$ la parametrización de un individuo de una población en un algoritmo evolutivo donde para todo i , x_i es un elemento del conjunto de parámetros \mathcal{P} .*

Llamaremos esquema a toda lista de caracteres sobre el conjunto de parámetros extendido $\mathcal{P} \cup \{\}$ donde el carácter $*$ denota el carácter comodín. Es decir, un esquema e es una lista de caracteres $e := (\varepsilon_1, \dots, \varepsilon_n)$ con $\varepsilon_i \in \mathcal{P} \cup \{*\}$ para todo i , $1 \leq i \leq n$.*

Nota 0.0.38. *Nótese que un esquema es la anti-imagen de una proyección y por lo tanto representa a un conjunto de individuos.*

Teorema 0.0.39 (Teorema de los Esquemas de Holland [Hol92]). *Bajo la acción de los operadores genéticos usuales los esquemas más simples, pero con mejor calificación media, tienden a reproducirse rápidamente.*

Nota 0.0.40. *Nótese que este teorema es un resultado asintótico que garantiza la convergencia hacia un extremo local, lo cual no es generalmente el resultado deseado. Este fenómeno es conocido como convergencia prematura y es uno de los principales problemas de los algoritmos evolutivos y la programación genética. Para mitigar este problema se suelen proporcionar mecanismos adicionales que garanticen la diversidad de la población.*

Ahora bien, dicha fundamentación también nos trae un resultado sorprendente.

Teorema 0.0.41 (No Free Lunch Theorem [WM97]). *Sean a y b dos algoritmos cualesquiera de optimización y \mathcal{F} el espacio de funciones sobre los que operan dichos algoritmos. Entonces para toda función a optimizar f_a existe otra f_b tal que el rendimiento del algoritmo a en f_a es igual al rendimiento de b en f_b .*

En otras palabras, todas las estrategias de tipo heurístico tienen el mismo rendimiento cuando se hace el estudio en media sobre el espacio de *todos* los posibles problemas. Esto *no* es equivalente a decir que todos los métodos funcionan igual para cualquier problema. Lo que afirma es que no existe un método de tipo heurístico cuyo comportamiento en promedio sobre todas las funciones sea mejor que otro. Es por esto que, ante un problema en particular, se requiera un estudio exhaustivo con el fin de averiguar con que método se obtienen los mejores resultados.

El problema de regresión simbólica

Como ya se ha mencionado en el Teorema 0.0.41 en general no existe un método superior al resto, pero esto no excluye que para problemas particulares unos métodos sean más indicados que otros. Esto justifica realizar algunas pruebas comparativas entre distintos métodos. Para ellos consideramos el problema de la regresión simbólica:

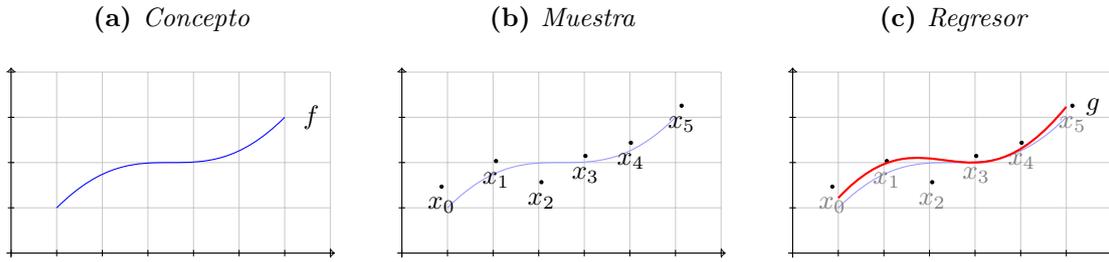
Definición 0.0.42. *Supongamos que se han obtenido N observaciones de un fenómeno cualquiera que pueda ser representado como una aplicación f de \mathbb{R}^n en \mathbb{R}^m . Se pide encontrar una aplicación g , dentro de un espacio de búsqueda \mathcal{F} , que mejor aproxime la aplicación f .*

Nota 0.0.43. *Generalmente estos problemas se encuadran dentro de la Teoría del Aprendizaje. En este contexto usualmente se denota por f al concepto a aprender, denotaremos por \mathcal{F} al espacio de búsqueda o de modelos y finalmente denotaremos por g al regresor. Usaremos esta nomenclatura a partir de ahora.*

En la Figura 2 podemos ver un ejemplo. La Subfigura (a) representa el concepto f . A partir de dicho concepto, de algún modo, obtenemos una muestra x_i , $0 \leq i \leq 5$ que podemos observar en la Subfigura (b). Sobre dicha muestra aplicamos el procedimiento de aprendizaje y obtenemos un regresor g que la aproxima siguiendo un modelo que hemos prefijado. En el caso que el procedimiento haya sido exitoso, diremos que el regresor ha *aprendido* el concepto original.

En este caso, el procedimiento no ha sido completamente exitoso pero, como podemos observar en la Subfigura (c), el regresor g se aproxima al concepto f .

Figura 2: Ejemplo de problema de regresión simbólica.



Hay que tener en cuenta las siguientes consideraciones:

- Las observaciones de que se disponen pueden contener errores.
- El espacio de búsqueda deberá ser *finitamente parametrizado*.

En el caso que el espacio de búsqueda o modelo sea un espacio vectorial normado los métodos de interpolación y/o mínimos cuadrados son óptimos, pues dan la mejor solución y su coste es muy pequeño. Sin embargo, en multitud de ocasiones el modelo puede no ser un espacio vectorial. En estos casos se puede intentar resolver el problema transformándolo en uno de *optimización no lineal*. Desgraciadamente ya hemos mencionado que las técnicas para resolver dichos problemas distan de ser eficaces. De todas formas el problema, en general, es mucho más complicado pues se desconoce el modelo por completo. Es aquí donde intervienen los algoritmos bioinspirados.

Es bien conocido que el mejor algoritmo bioinspirado que se dispone para enfrentarse a este problema es la programación genética (véase por ejemplo [Koz92]). Recordamos que para poder usar este método debemos presentar: una población, un mecanismo de evolución, un mecanismo de calificación y una condición de parada.

El mecanismo de calificación es extremadamente simple en esta problema. Simplemente basta calcular el error cuadrático medio (*MSE*) (véase la Sección 8.2 para más detalles) y en el caso que nuestra muestra contenga ruido, penalizarlo de acuerdo a ciertos parámetros. Igualmente sencilla es la condición de parada.

Parar cuando alguna de las siguientes condiciones se cumpla:

- El *MSE* sea inferior a un cierto valor aceptable.
- El número de operaciones supere un cierto límite.

Veamos ahora la población y los mecanismos de evolución.

Estructura de datos usada para la población

Nuestro objetivo es encontrar una aplicación dentro de un espacio de búsqueda finitamente parametrizado. Clásicamente la estructura de datos usada para realizar dicha

tomaremos como hijo el elemento de posición $i + 1$ siempre que $i + 1 \leq L$. En este último caso tomaríamos un elemento del conjunto de terminales de forma aleatoria.

En general, para generar el elemento en la posición k , $2 \leq k \leq L$ tomamos aleatoriamente una función $f_{j_k} \in \mathcal{F}$ y para cada argumento $i \in \{1, \dots, a_{j_k}\}$ de esta función f_{j_k} escogemos aleatoriamente un elemento α_i del conjunto $\mathcal{T} \cup \{u_{j+1}, \dots, u_L\}$. Como hemos hecho previamente, si α_i pertenece al conjunto de terminales introducimos ese elemento como hijo. En otro caso tomaremos como hijo el elemento de posición $A(j - 1) + i + 1$ siempre y cuando $A(j - 1) + i + 1 \leq L$. En este último caso, como hicimos anteriormente, tomaríamos un elemento del conjunto de terminales de forma aleatoria.

Nótese que los árboles generados de esta forma pueden contener nodos que nunca serán computados. De esta forma conseguimos que la longitud de todos los árboles de la población sea uniforme. En la Sección 8.1 comentaremos más en detalle este aspecto. Nótese que este tipo de homogeneidad no es posible con poblaciones de árboles siguiendo las estructuras clásicas.

Además de poder generar una población es necesario definir los operadores genéticos. Como norma general, los operadores de selección y reproducción no dependen de la estructura de datos usada. En la Sección 8.2.3 entraremos más en detalle sobre ellos. Sin embargo, los operadores de cruce y mutación si que dependen fuertemente de la estructura de datos usada. Pasamos a detallar los más usados.

Operador de mutación árbol: Se escoge un nodo al azar y se modifica la función asociado a dicho nodo.

Operador de cruce árbol: Se escoge un nodo de cada uno de los elementos a cruzar y se intercambian las *subexpresiones* que definen cada uno de los nodos. En la Figura 4 se puede ver un ejemplo de este operador.

Nuestra principal contribución al respecto se encuentra en el Capítulo 8.1 (véase también [AMPB09]) donde introduciremos de forma novedosa la estructura de datos *slp's*, vista anteriormente, en este contexto y que representa una gran mejora respecto de los árboles. Además desarrollamos los mecanismos evolutivos adaptados a dicha estructura de datos y presentamos un estudio empírico entre ambas estructuras de datos de las que se concluye que los *slp's* tiene un rendimiento superior a los árboles para resolver instancias del problema de regresión simbólica.

La elección del modelo

Ya hemos visto que la programación genética se aplica cuando se desconoce por completo el modelo. En general, el Álgebra Lineal nos provee de un modelo trivial: el polinomio interpolador. Sin embargo es bien sabido que en casi ningún caso es el modelo adecuado pues Runge mostró en [Run01] que la diferencia entre el polinomio interpolador y el concepto es muy grande lejos de los puntos de interpolación, sobre todo cuando el grado del polinomio es alto. Para mitigar este problema se han desarrollado múltiples técnicas que intentan disminuir el grado de polinomio. Entre ellas podemos enunciar *mínimos cuadrados e interpolación vía splines*.

En general, este es un fenómeno conocido desde el siglo XIV como *navaja de Ockham*: «Siempre se debe preferir un modelo simple a uno complejo». En nuestro problema no es muy complicado ver que el grado del polinomio interpolador es una buena medida de la *complejidad* del modelo.

Ahora bien, los conceptos a aprender pueden llegar a ser realmente complejos sin necesidad de buscar ejemplos «patológicos». Luego es necesario un balance entre la complejidad del modelo y lo bien que aproxima el concepto.

Entre los métodos propuestos para resolver este problema se encuentran los métodos basados en propiedades estadísticas de los modelos. Entre ellos cabe destacar el criterio de información de Akaike (*AIC*) y el criterio de información Bayesiano (*BIC*) (véase [Aka74] y [Sch78]). Por otro lado, otra forma de afrontar el problema es a partir de la capacidad de *clasificación* que presenta el modelo que en este contexto se denomina dimensión de Vapnik-Chervonenkis (VCD). Este enfoque fue desarrollado por Vapnik y Chervonenkis en sus trabajos [VC74, Vap98, VC71].

Nuestra principal contribución al respecto se encuentra en el Capítulo 9 (véase también [MABC09]) en el que calculamos la dimensión de Vapnik-Chervonenkis (VCD) de los *slp's* (Teorema 9.0.1) y usamos dicho valor para crear un mecanismo de selección del modelo adecuado (véase [BAM10]) basado en los principios de minimización del riesgo estructural (*SRM*). Además realizamos un estudio empírico en el que demostramos que esta forma de seleccionar los modelos muestra un comportamiento superior a los basados en propiedades estadísticas.

Optimización de las constantes del modelo modelo

Una vez obtenido el modelo f es posible que sea necesario ajustar ciertas constantes. Tradicionalmente el proceso para ajustar las constantes del modelo se basa en encontrar un *estimador de máxima verosimilitud* para el parámetro en concreto. Sin embargo esta aproximación puede llegar a ser muy complicada. Por otro lado, este problema se puede considerar como *otro* problema de optimización.

Problema 0.0.44. *Dada una muestra, encontrar el vector de constantes c pertenecientes al conjunto C tal que minimice el error del modelo sobre la muestra (x_i, y_i) , $1 \leq i \leq m$. Es decir, buscar las constantes c tales que:*

$$\min_{c \in C} \sum_{i=1}^m \|f_c(x_i) - y_i\|,$$

donde f_c denota el modelo resultante de sustituir el valor numérico de las constantes c en el modelo f .

Como ya hemos visto podemos afrontar este problema desde un frente clásico usando, por ejemplo, técnicas de descenso de gradiente o bien con una estrategia evolutiva. En el Capítulo 10 (véase también [AMB09] y [BADIP⁺10]) presentamos diversos algoritmos co-evolutivos para afrontar dicho problema. La idea es ir ajustando las constantes *a la vez* que vamos construyendo el modelo. También presentamos un estudio empírico en el

que comparamos los distintos métodos en el que no encontramos diferencias significativas entre los distintos métodos propuestos.

Una solución bioinspirada al problema XVII de Smale

Para finalizar con esta introducción comentaremos la mayor contribución de esta memoria: una serie de algoritmos evolutivos que resuelven, en la práctica, el problema XVII de Smale en el caso real. Estos algoritmos se basan en la búsqueda de ceros aproximados directamente. Como hemos visto anteriormente, para que un punto x_0 sea un cero aproximado de un sistema de ecuaciones f es necesario que $\alpha(f, x_0) < \alpha_0$ (véase el Teorema 0.0.8). Por lo tanto, el problema XVII de Smale quedaría redactado, en este contexto, de la siguiente manera:

Problema 0.0.45 (XVII de Smale, caso α^*). *Sea $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ un sistema de ecuaciones polinomiales multivariado, encontrar un punto x_0 tal que $\alpha(f, x_0) < \alpha_0$.*

Sin embargo, calcular α mediante su definición es una operación muy costosa desde un punto de vista computacional (véase la discusión al respecto en el Capítulo 12). Es por ello que introducimos la siguiente magnitud:

Definición 0.0.46 (α^*). *Usando la notación anterior. Definimos $\alpha^*(f, x)$ como el siguiente producto:*

$$\alpha^*(f, x) := \beta(f, x) \mu_{norm}(f, x) = \frac{D^{3/2}}{2} \|D_x(f)^{-1} f(x)\| \| \|x\| D_x(f)^{-1} \Delta((d)^{1/2}) \| ,$$

donde $\beta(f, x)$ fue definida en la Definición 0.0.5 y $\mu_{norm}(f, x)$ representa el condicionamiento normalizado.

A partir de dicha definición obtenemos un test probabilístico para la propiedad «ser cero aproximado» que se basa en el siguiente teorema:

Teorema 0.0.47 (Test cero aproximado). *Usando la notación anterior. Sea ε un número real tal que $0 < \varepsilon \leq \frac{1}{2}$, $f \in \mathcal{P}_{(d)}^{\mathbb{R}}$ un sistema de ecuaciones polinomiales multivariadas y $x_0 \in \mathbb{R}^n$ un punto afín con norma acotada por*

$$\|x_0\| \leq \frac{\Gamma(1/4)^2 \mathcal{D}}{\varepsilon^2}.$$

Bajo estas hipótesis, x_0 es un cero aproximado de f con probabilidad mayor que $1 - \varepsilon$, si y solo si se cumple alguna de las siguientes condiciones:

1. $\alpha^*(f, x_0) \leq \alpha_0$.
2. $\alpha^*(f, x_k) \leq \alpha_0$ donde x_k denota una iteración de Newton con

$$k \in \mathcal{O} \left(\log \log \left(\frac{(Dn)^{3/2} N^{1/4} \mathcal{D}^{5/4}}{\varepsilon^{5/2}} \right) \right),$$

donde D denota el máximo de la lista de grados (d) , N denota la dimensión como espacio vectorial de $\mathcal{P}_{(d)}^{\mathbb{R}}$, \mathcal{D} denota el número de Bézout y α_0 es la constante definida en el Teorema 0.0.8.

El número de operaciones necesario para aplicar dicho test está acotado polinomialmente por lo que podemos construir varios algoritmos evolutivos que encuentran uno o varios ceros aproximados. La idea es minimizar la función $\alpha^*(f, x)$ con la esperanza de encontrar un cero aproximado, es decir, un punto x tal que:

$$\alpha^*(f, x) < \alpha_0.$$

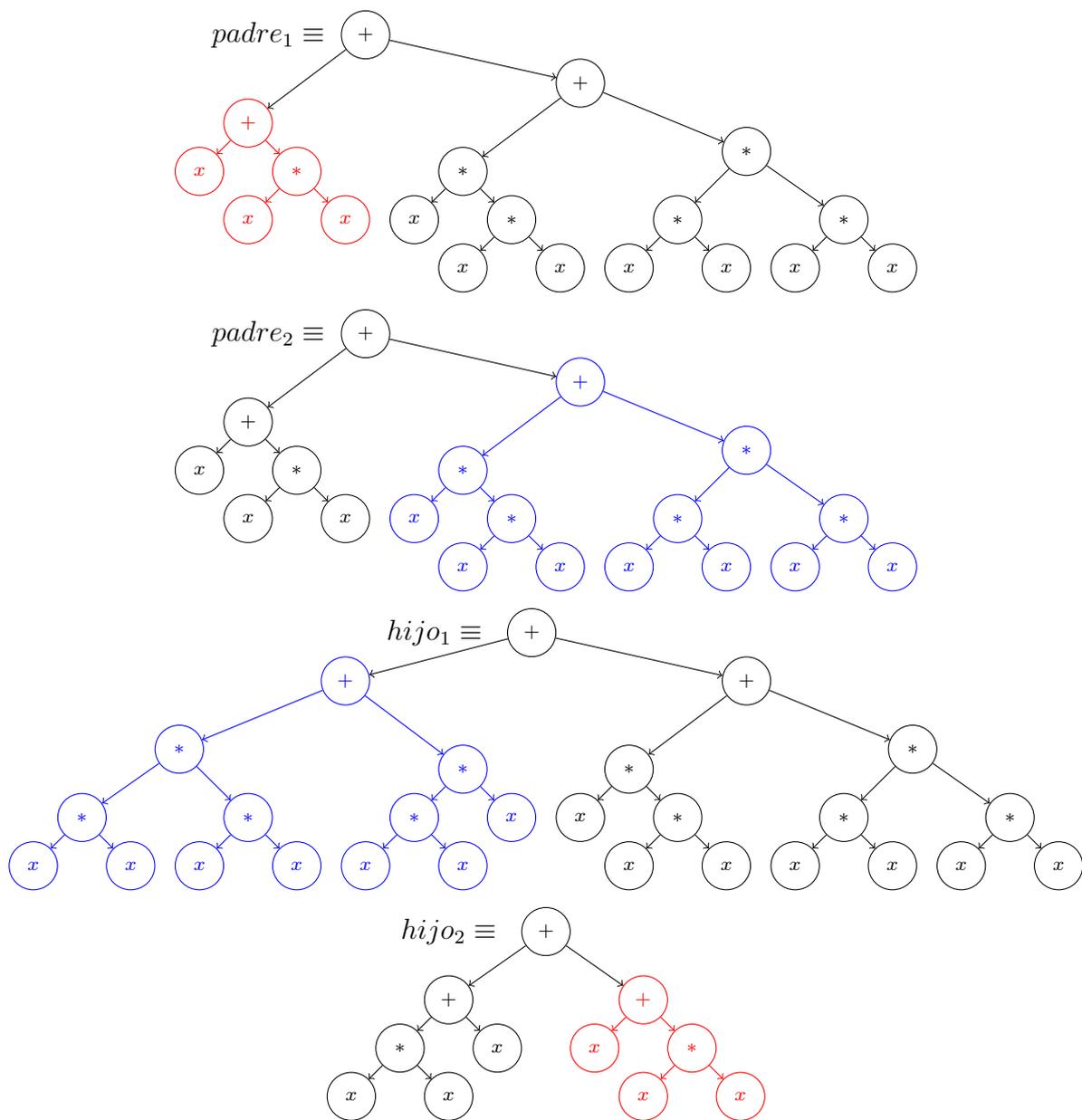
De esta manera, tendremos que fomentar la exploración de las zonas donde el valor de $\alpha^*(f, x)$ sea bajo, pero sin descuidar otras zonas. Para esta tarea, los algoritmos evolutivos se han demostrado estar muy capacitados.

En el Capítulo 12 propondremos varios algoritmos evolutivos siguiendo las siguientes estrategias:

- Algoritmo evolutivo genético.
- Evolución diferencial.
- Enfriamiento simulado.
- Enjambre de partículas.

Además, en la Sección 12.4, presentaremos resultados empíricos que demuestran que estos algoritmos son capaces de encontrar ceros aproximados de sistemas de 10 incógnitas con relativa facilidad.

Figura 4: Árboles hijos mediante el cruce de arboles



Parte I

La construcción de un algoritmo numérico de deformación homotópica

Capítulo 1

Notaciones generales

*Young man, in mathematics you don't understand things. You just get used to them*¹.
Atribuido a **John von Neumann** [Zuk84, pág 208]

Dedicaremos las siguientes páginas a enunciar las propiedades básicas, con sus referencias, que usaremos a lo largo del manuscrito. Así pues:

Sean $n, d \in \mathbb{N}$ dos números naturales. Denotamos por $P_d^{\mathbb{R}}$ al espacio vectorial de todos los polinomios $f \in \mathbb{R}[X_1, \dots, X_n]$ con coeficientes en los números reales y grado menor que d , es decir $\deg(f) \leq d$. Así mismo, denotamos por $H_d^{\mathbb{R}}$ al espacio vectorial de todos los polinomios homogéneos $g \in \mathbb{R}[X_0, \dots, X_n]$ con coeficientes en los números reales y de grado d . Para todo número $m \in \mathbb{N}$ con $m \leq n$ y para toda lista de grados $(d) := (d_1, \dots, d_m) \in \mathbb{N}^m$, denotamos por $\mathcal{P}_{(d)}^{\mathbb{R}}$ al espacio vectorial de los sistemas de ecuaciones polinomiales multivariadas de grados menor que (d) y coeficientes en los números reales. Es decir,

$$\mathcal{P}_{(d)}^{\mathbb{R}} := P_{d_1}^{\mathbb{R}} \times \dots \times P_{d_m}^{\mathbb{R}}.$$

Igualmente, denotaremos por $\mathcal{H}_{(d)}^{\mathbb{R}}$ al espacio vectorial de los sistema de ecuaciones polinomiales homogéneas multivariadas de grados (d) y coeficientes en los números reales. Es decir,

$$\mathcal{H}_{(d)}^{\mathbb{R}} := H_{d_1}^{\mathbb{R}} \times \dots \times H_{d_m}^{\mathbb{R}}.$$

Nótese que tanto $\mathcal{P}_{(d)}^{\mathbb{R}}$ como $\mathcal{H}_{(d)}^{\mathbb{R}}$ son dos espacios vectoriales de la misma dimensión (de hecho son isomorfos):

$$N := m + \sum_{i=1}^m \binom{d_i + n}{n}.$$

Para toda lista $f := (f_1, \dots, f_m) \in \mathcal{P}_{(d)}^{\mathbb{R}}$ de polinomios denotamos por $V_{\mathbb{A}}(f) \subseteq \mathbb{R}^n$ a la variedad algebraica afín de los ceros comunes de f en \mathbb{R}^n . Es decir:

$$V_{\mathbb{A}}(f) := \{x \in \mathbb{R}^n : f_i(x) = 0 \forall i, 1 \leq i \leq m\}.$$

¹Joven, en matemáticas uno no entiende cosas. Simplemente te acostumbras a ellas.

Así mismo para toda lista $f := (f_1, \dots, f_m) \in \mathcal{H}_{(d)}^{\mathbb{R}}$ de polinomios homogéneos, denotaremos por $V_{\mathbb{P}}(f) \subseteq \mathbb{P}_n(\mathbb{R})$ a la variedad algebraica proyectiva de los ceros comunes de f en $\mathbb{P}_n(\mathbb{R})$, donde $\mathbb{P}_n(\mathbb{R})$ denota el espacio proyectivo real de dimensión n . Es decir:

$$V_{\mathbb{P}}(f) := \{x \in \mathbb{P}_n(\mathbb{R}) : f_i(x) = 0 \forall i, 1 \leq i \leq m\}.$$

Así mismo denotaremos por $\varphi_i : \mathbb{R}^n \rightarrow \mathbb{P}_n(\mathbb{R})$ a la aplicación afín dada por:

$$\varphi_i(x_1, \dots, x_i, \dots, x_n) := (x_0 : x_1 : \dots : 1 : \dots : x_n), \quad \forall (x_1, \dots, x_n) \in \mathbb{R}^n,$$

donde $(x_0 : x_1 : \dots : x_n)$ denota coordenadas homogéneas. Nótese que:

$$V_{\mathbb{A}}(f) := \varphi_0^{-1}(V_{\mathbb{P}}(f)).$$

Definimos ahora:

- La variedad de incidencia real

$$V^{\mathbb{R}} \subseteq \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \times \mathbb{P}_n(\mathbb{R}),$$

como la variedad multi-homogénea dada por la siguiente identidad:

$$V^{\mathbb{R}} := \{(f, x) \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \times \mathbb{P}_n(\mathbb{R}) : f(x) = 0\}.$$

- La variedad de incidencia mixta

$$V^{\mathbb{R}, \mathbb{C}} \subseteq \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \times \mathbb{P}_n(\mathbb{C}),$$

como la variedad multi-homogénea dada por la siguiente identidad:

$$V^{\mathbb{R}, \mathbb{C}} := \{(f, x) \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \times \mathbb{P}_n(\mathbb{C}) : f(x) = 0\}.$$

1.1 Normas

Pasamos ahora a introducir el producto escalar de Bombieri (usado también por Weyl, Kostlan y otros) en $\mathcal{H}_{(d)}^{\mathbb{R}}$. Existe una amplia bibliografía sobre este producto escalar. Podemos citar, entre otros, [BBEM90], [Kos93], [Ded96], [Dég01], [MR04] o [BCSS98].

Siguiendo el convenio anterior, sea $d \in \mathbb{N}$ un número natural y sea $H_d^{\mathbb{R}} \subseteq \mathbb{R}[X_0, \dots, X_n]$ el espacio vectorial de todos los polinomios homogéneos de grado d con coeficientes en los números reales. La base monomial en $H_d^{\mathbb{R}}$ se suele identificar con el conjunto de multi-índices:

$$\mathbb{N}_d^{n+1} := \{\mu = (\mu_0, \mu_1, \dots, \mu_n) \in \mathbb{N}^{n+1} : |\mu| = \mu_0 + \dots + \mu_n = d\}.$$

Como tradicionalmente se realiza en *Álgebra Conmutativa Computacional*, podemos escoger un orden monomial \leq_d en \mathbb{N}_d^{n+1} (para más información consultar [CLO97], [BW93] o [GPW03] que contienen una introducción a los *órdenes monomiales*, a las *bases de Gröbner* y al *Álgebra Computacional Conmutativa*). Este orden monomial en \mathbb{N}_d^{n+1} nos permite

ver los elementos de $H_d^{\mathbb{R}}$ como vectores dados por sus coordenadas (respecto del orden monomial). A esta forma de codificar un polinomio se la suele denominar *codificación densa* en la literatura clásica de *Geometría Algebraica Efectiva*.

Para cada $\mu \in \mathbb{N}_d^{n+1}$, definimos el coeficiente multinomial como:

$$\binom{d}{\mu} := \frac{d!}{\mu_0! \cdots \mu_n!}.$$

Denotaremos por $\Delta_d \in \mathcal{M}_{N_d}(\mathbb{R})$, donde $N_d := \binom{d+n}{n}$, a la matriz diagonal asociada a $H_d^{\mathbb{R}}$ cuya entrada μ -ésima (con respecto al orden monomial \leq_d) en la diagonal es $\binom{d}{\mu}^{-1/2}$. Es decir, Δ_d es la matriz diagonal:

$$\Delta_d := \bigoplus_{\mu \in \mathbb{N}_d^{n+1}} \binom{d}{\mu}^{-1/2}.$$

Sea $\langle \cdot, \cdot \rangle_d : H_d^{\mathbb{R}} \times H_d^{\mathbb{R}} \rightarrow \mathbb{R}$ el producto escalar euclídeo en el espacio $H_d^{\mathbb{R}}$ y sea $(d) := (d_1, \dots, d_m) \in \mathbb{N}^m$ una lista de números naturales (que usualmente llamaremos *lista de grados*). Podemos extender de forma natural este producto escalar a $\mathcal{H}_{(d)}^{\mathbb{R}}$ usando la siguiente identidad:

$$\langle f, g \rangle := \sum_{i=1}^m \langle f_i, g_i \rangle_{d_i} \in \mathbb{R},$$

donde $f = (f_1, \dots, f_m)$, $g = (g_1, \dots, g_m) \in \mathcal{H}_{(d)}^{\mathbb{R}}$. Luego, denotamos por $\langle \cdot, \cdot \rangle_{\Delta}$ al producto escalar de Bombieri en $\mathcal{H}_{(d)}^{\mathbb{R}}$ definido por las respectivas matrices Δ_{d_i} . Es decir:

$$\langle f, g \rangle_{\Delta} := \sum_{i=1}^m \langle \Delta_{d_i} f_i, \Delta_{d_i} g_i \rangle_{d_i} = \sum_{i=1}^m \langle f_i, g_i \rangle_{\Delta_{d_i}}$$

donde $f := (f_1, \dots, f_m)$, $g := (g_1, \dots, g_m) \in \mathcal{H}_{(d)}^{\mathbb{R}}$.

Así mismo, denotaremos por $\|\cdot\|_{\Delta}$ a la norma definida en $\mathcal{H}_{(d)}^{\mathbb{R}}$ por el producto escalar de Bombieri $\langle \cdot, \cdot \rangle_{\Delta}$ y finalmente, llamaremos matriz de Kostlan a la siguiente matriz Δ :

$$\Delta := \bigoplus_{i=1}^m \Delta_{d_i} \in \mathcal{M}_N(\mathbb{R}).$$

Denotaremos por $\mathcal{O}(k, \mathbb{R}) \subseteq \mathcal{M}_k(\mathbb{R})$ al grupo ortogonal. A lo largo de esta memoria usaremos la siguiente acción del grupo $\mathcal{O}(n+1, \mathbb{R})$ sobre $\mathcal{H}_{(d)}^{\mathbb{R}}$:

$$\begin{aligned} \mathcal{O}(n+1, \mathbb{R}) \times \mathcal{H}_{(d)}^{\mathbb{R}} &\longrightarrow \mathcal{H}_{(d)}^{\mathbb{R}} \\ (U, f) &\longmapsto f \circ U, \end{aligned}$$

donde \circ denota la composición. Esta acción se extiende de forma natural al espacio proyectivo de sistemas homogéneos $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$. Entonces podemos asegurar el siguiente resultado (cf. [BCSS98]):

Proposición 1.1.1 (Invarianza Ortogonal). *La aplicación bilineal $\langle \cdot, \cdot \rangle_\Delta$ es invariante respecto a la acción del grupo ortogonal. Esto es, sea $f, g \in \mathcal{H}_{(d)}^\mathbb{R}$ y $U \in \mathcal{O}(n+1, \mathbb{R})$ entonces:*

$$\langle f \circ U, g \circ U \rangle_\Delta = \langle f, g \rangle_\Delta.$$

El producto escalar de Bombieri $\langle \cdot, \cdot \rangle_\Delta$ induce una estructura de variedad de Riemann en el espacio proyectivo de sistemas homogéneos $\mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R})$. Para cada punto $f \in \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R})$ denotamos por f^\perp el complemento ortogonal respecto a este producto escalar. Nótese que el espacio tangente $T_f \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R})$ se puede identificar con el complemento ortogonal f^\perp con respecto al producto escalar heredado de $\mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R})$. Asimismo, la carta afín en $f \in \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R})$ es la aplicación:

$$\begin{aligned} \varphi_{\tilde{f}} : f^\perp &\longrightarrow \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R}) \setminus f^\perp, \\ g &\longmapsto \tilde{f} + g, \end{aligned}$$

donde $\tilde{f}, \tilde{g} \in \mathcal{H}_{(d)}^\mathbb{R}$ son representantes de f y g de norma 1. Además, la aplicación tangente $T_0 \varphi_f : f^\perp \longrightarrow T_f \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R})$ es una isometría.

Ahora bien, asumimos que el espacio proyectivo $\mathbb{P}_n(\mathbb{R})$ está dotado de la estructura de variedad de Riemann canónica. A lo largo de esta memoria usaremos la siguiente acción del del grupo ortogonal $\mathcal{O}(n, \mathbb{R})$ sobre \mathbb{R}^n :

$$\begin{aligned} \mathcal{O}(n, \mathbb{R}) \times \mathbb{R}^n &\longrightarrow \mathbb{R}^n \\ (U, x) &\longmapsto U^{-1}x, \end{aligned}$$

Esta acción se extiende de forma natural a $\mathbb{P}_n(\mathbb{R})$. Trivialmente se tiene entonces:

Proposición 1.1.2 (Invarianza Ortogonal). *La aplicación bilineal $\langle \cdot, \cdot \rangle_2$ es invariante respecto a la acción del grupo ortogonal. Esto es, sea $x, y \in \mathbb{P}_n(\mathbb{R})$ y $U \in \mathcal{O}(n+1, \mathbb{R})$ entonces:*

$$\langle U^{-1}x, U^{-1}y \rangle_2 = \langle x, y \rangle_2.$$

Para cada punto $x \in \mathbb{P}_n(\mathbb{R})$ denotamos por x^\perp el complemento ortogonal respecto a este producto escalar. Nótese que el espacio tangente $T_x \mathbb{P}_n(\mathbb{R})$ se puede identificar con el complemento ortogonal x^\perp con respecto al producto escalar heredado de $\mathbb{P}_n(\mathbb{R})$. Asimismo, la carta afín en $x \in \mathbb{P}_n(\mathbb{R})$ es la aplicación:

$$\begin{aligned} \varphi_{\tilde{x}} : x^\perp &\longrightarrow \mathbb{P}_n(\mathbb{R}) \setminus x^\perp, \\ y &\longmapsto \tilde{x} + y \end{aligned}$$

donde $\tilde{x}, \tilde{y} \in \mathbb{R}^n$ son representantes de x y y de norma 1. Además, la aplicación tangente $T_0 \varphi_x : x^\perp \longrightarrow T_x \mathbb{P}_n(\mathbb{R})$ es una isometría.

Dado un par $(f, x) \in \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R}) \times \mathbb{P}_n(\mathbb{R})$, denotamos por $T_x f := (d_x f)|_{x^\perp}$ a la restricción de la aplicación tangente $d_x f$ al espacio tangente x^\perp , donde el par (f, x) es cualquier representación afín tal que $\|f\|_\Delta = \|x\|_2 = 1$. De vez en cuando identificaremos $T_x f$ y la matriz Jacobiana en cualquier base ortogonal de x^\perp . En el caso de que $x = e_0 := (1 : 0 : \dots : 0)$, identificaremos:

$$T_{e_0} f \equiv \begin{pmatrix} \frac{\partial f^1}{\partial x_1}(e_0) & \dots & \frac{\partial f^1}{\partial x_n}(e_0) \\ \vdots & & \vdots \\ \frac{\partial f^m}{\partial x_1}(e_0) & \dots & \frac{\partial f^m}{\partial x_n}(e_0) \end{pmatrix},$$

para cualquier representación de $f \in \mathcal{H}_{(d)}^{\mathbb{R}}$ con $\|f\|_{\Delta} = 1$.

1.2 La variedad de incidencia, la variedad singular y el triángulo de proyecciones

Previamente hemos definido la variedad de incidencia real $V^{\mathbb{R}} \subseteq \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \times \mathbb{P}_n(\mathbb{R})$ como la variedad multi-homogénea dada por la siguiente igualdad

$$V^{\mathbb{R}} := \{(f, x) \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \times \mathbb{P}_n(\mathbb{R}) : f(x) = 0\}.$$

A partir de las dos isometrías definidas en la Subsección 1.1 queda definida una acción del grupo ortogonal $\mathcal{O}(n+1, \mathbb{R})$ sobre $V^{\mathbb{R}}$ tal que para cada $U \in \mathcal{O}(n+1, \mathbb{R})$ la siguiente es una isometría en $V^{\mathbb{R}}$:

$$\begin{aligned} V^{\mathbb{R}} &\longrightarrow V^{\mathbb{R}}, \\ (f, \zeta) &\mapsto (f \circ U, U^{-1}\zeta) \end{aligned}$$

Sean π_1 y π_2 las proyecciones canónicas desde $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \times \mathbb{P}_n(\mathbb{R})$ a $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ y $\mathbb{P}_n(\mathbb{R})$ respectivamente. Abusando de la notación, denotaremos por π_1 y π_2 a sus restricciones sobre $V^{\mathbb{R}}$ obteniendo el siguiente *triángulo de proyecciones*:

$$\begin{array}{ccc} & V^{\mathbb{R}} & \\ \pi_1 \swarrow & & \searrow \pi_2 \\ \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) & & \mathbb{P}_n(\mathbb{R}). \end{array}$$

Se tiene entonces que:

Proposición 1.2.1. *La proyección π_2 es una submersión para todo punto $(f, \zeta) \in V^{\mathbb{R}}$. Es decir, el conjunto de puntos singulares y valores críticos de π_2 es vacío. Además, las fibras de π_2 en cualquier punto forman una subvariedad lineal dada por la identidad:*

$$V_x^{\mathbb{R}} := \pi_2^{-1}(x) \cong \{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) : f(x) = 0\}.$$

Es más, como la codimensión de $V_x^{\mathbb{R}}$ en $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ es m , tenemos que su dimensión es

$$\dim(V_x^{\mathbb{R}}) = N - m.$$

Por último, para cualquier matriz ortogonal $U \in \mathcal{O}(n+1, \mathbb{R})$, dados $x, y \in \mathbb{P}_n(\mathbb{R})$ tales que $Ux = y$, se tiene que la siguiente expresión:

$$\begin{aligned} V_x^{\mathbb{R}} &\longrightarrow V_y^{\mathbb{R}} \\ f &\longrightarrow f \circ U^{-1}, \end{aligned}$$

es una isometría entre las fibras $V_x^{\mathbb{R}}$ y $V_y^{\mathbb{R}}$.

La aplicación π_1 por lo general no es suprayectiva. Llamaremos sistemas factibles a la imagen de π_1 y lo denotaremos por

$$\mathcal{R}_{(d)} := \pi_1(V) \subseteq \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}).$$

La siguiente proposición es consecuencia inmediata de diversos resultados de *Teoría de la Eliminación* referidos a los números reales.

Proposición 1.2.2. *El conjunto $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \mathcal{R}_{(d)}$ tiene medida nula si y sólo si todos los grados d_1, \dots, d_m son números impares.*

Denotaremos por $\Sigma'^{\mathbb{R}} \subseteq V$ al conjunto de los puntos singulares de la proyección π_1 y por $\Sigma^{\mathbb{R}}$ para el conjunto de los valores críticos de π_1 que pertenezcan a $\mathcal{R}_{(d)}$. Llamaremos a este último conjunto variedad discriminante.

La siguiente igualdad caracteriza los puntos regulares de π_1 en $V^{\mathbb{R}}$:

$$V \setminus \Sigma'^{\mathbb{R}} = \{(f, x) \in V : T_x f \text{ tiene rango } m\}.$$

Estudiamos ahora en profundidad este conjunto.

Proposición 1.2.3. *Usando la notación anterior:*

- $\pi_1 : V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}} \rightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma^{\mathbb{R}}$ es una submersión en todo punto $(f, x) \in V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$.
- Las dimensiones de $\Sigma'^{\mathbb{R}}$ y $\Sigma^{\mathbb{R}}$ sobre los números reales son:

$$\begin{aligned} \dim(\Sigma'^{\mathbb{R}}) &\leq N + n - m - 1 \\ \dim(\Sigma^{\mathbb{R}}) &= N - 1. \end{aligned}$$

- Si $m < n$ el conjunto abierto $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma^{\mathbb{R}}$ tiene una única componente conexa.
- Para todo $f \in \mathcal{R}_{(d)} \setminus \Sigma^{\mathbb{R}}$ la fibra $\pi_1^{-1}(f)$ es una variedad algebraica proyectiva de dimensión $n - m$.

Consideramos ahora la siguiente aplicación continua que surge de la acción del grupo ortogonal $\mathcal{O}(n + 1, \mathbb{R})$ sobre $V^{\mathbb{R}}$ tal que para cada $U \in \mathcal{O}(n + 1, \mathbb{R})$ la siguiente aplicación es una isometría en $V^{\mathbb{R}}$:

$$\begin{aligned} \Phi : \mathcal{O}(n + 1, \mathbb{R}) \times V^{\mathbb{R}} &\longrightarrow V^{\mathbb{R}} \\ (U, (f, x)) &\longmapsto (f \circ U, U^{-1}x), \end{aligned} \tag{1.1}$$

Lema 1.2.4. *Usando la notación anterior. La aplicación continua Φ cumple:*

- Φ es sobre.
- Para toda $U \in \mathcal{O}(n + 1, \mathbb{R})$, sea $y = Ux$ entonces Φ define un isometría entre $V_x^{\mathbb{R}}$ y $V_y^{\mathbb{R}}$. Es decir, la siguiente aplicación es una isometría entre espacios euclídeos:

$$\begin{aligned} U : V_x^{\mathbb{R}} &\longrightarrow V_y^{\mathbb{R}} \\ (f, x) &\longmapsto (f \circ U, U^{-1}x) \end{aligned}$$

Nota 1.2.5. *El resultado anterior sigue siendo válido si restringimos la acción al grupo ortogonal especial $\mathrm{SO}(m, \mathbb{R})$.*

A partir del resultado anterior se deduce inmediatamente:

Corolario 1.2.6. *Usando la notación anterior.*

$$\mathrm{SO}(m, \mathbb{R}) \times \Phi((V_{e_0}^{\mathbb{R}} \setminus \Sigma^{\mathbb{R}}) = V^{\mathbb{R}} \setminus \Sigma^{\mathbb{R}}).$$

1.3 Algunas fórmulas integrales

En esta subsección introduciremos algunos resultados relevantes de *Geometría Integral* que usaremos a lo largo del documento. Para cada variedad de Riemann M de grado d y para cualquier conjunto medible $A \subseteq M$ denotaremos por $\nu_d[A]$ su medida de Hausdorff de dimensión d . Como ya hemos comentado anteriormente, denotaremos por $\mathbb{P}_k(\mathbb{R})$ al espacio proyectivo real de dimensión k y usaremos la notación $d\nu_k^{\mathbb{P}}$ para la forma de volumen canónica asociada como variedad de Riemann. Para cada subconjunto $A \subseteq \mathbb{P}_k(\mathbb{R})$, denotaremos su volumen, como subconjunto de dimensión k , por $\nu_k[A]$. Finalmente, denotaremos como ϑ_k al volumen de $\mathbb{P}_k(\mathbb{R})$. Es decir:

$$\vartheta_k := \nu_k[\mathbb{P}_k(\mathbb{R})] = \frac{\pi^{\frac{k}{2}}}{\Gamma(\frac{k}{2})}.$$

Sea $t > 0$ un número real positivo. Denotaremos por $\mathbb{S}_k^t(\mathbb{R}) \subseteq \mathbb{R}^{k+1}$ a la esfera de radio t y centrada en 0. Nótese que el volumen de $\mathbb{S}_k^t(\mathbb{R})$ como subvariedad de Riemann de \mathbb{R}^{k+1} es

$$\nu_k[\mathbb{S}_k^t(\mathbb{R})] := 2t^k \frac{\pi^{\frac{k}{2}}}{\Gamma(\frac{k}{2})} = 2t^k \vartheta_k.$$

Finalmente, denotaremos por $\mathbb{S}_k(\mathbb{R})$ a la esfera de radio $t = 1$.

Como en [SS93b] (cf. también [BCSS98]), consideraremos la estructura de variedad de Riemann en $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ inducida por el producto escalar euclídeo $\langle \cdot, \cdot \rangle$. Denotaremos por $d\nu_{\Delta}$ la forma de volumen de $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ asociada a esta estructura Riemannian.

Como $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ tiene volumen finito, podemos definir una distribución de probabilidad en $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$. Para toda función integrable $f : \mathbb{P}_n(\mathbb{R}) \rightarrow \mathbb{R}$ definiremos su esperanza $E_{\Delta}[f]$ como:

$$E_{\Delta}[f] := \frac{1}{\vartheta_N} \int_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} f d\nu_{\Delta}.$$

Nótese que $\vartheta_N = \nu_{\Delta}[\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})]$. Para cada subconjunto medible $A \subseteq \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$, definiremos la probabilidad del subconjunto como la esperanza de su función característica $\chi_A : \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \rightarrow \{0, 1\}$. Es decir:

$$\mathrm{Prob}_{\Delta}[A] := \frac{\nu_{\Delta}[A]}{\vartheta_N} = \frac{1}{\vartheta_N} \int_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \chi_A d\nu_{\Delta}.$$

Para todo subconjunto A de $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ denotaremos por $\tilde{A} \subseteq \mathcal{H}_{(d)}^{\mathbb{R}}$ al cono asociado a A . Es decir:

$$\tilde{A} := \pi^{-1}(A) \cup \{0\},$$

donde $\pi : \mathcal{H}_{(d)}^{\mathbb{R}} \setminus \{0\} \rightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ es la proyección canónica sobre el espacio proyectivo.

Como se puede ver en la Proposición 1 del Capítulo 11 de [BCSS98], la distribución de probabilidad definida por $d\nu_{\Delta}$ sobre $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ coincide con la *distribución normal* en $\mathcal{H}_{(d)}^{\mathbb{R}}$ respecto a la norma $\|\cdot\|_{\Delta}$. Es decir, para cada subconjunto medible $A \subset \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ se tiene que:

$$\text{Prob}_{\Delta}[A] = \frac{1}{(2\pi)^{N/2}} \int_{\mathcal{H}_{(d)}^{\mathbb{R}}} \chi_{\tilde{A}} e^{-\frac{\|f\|_{\Delta}^2}{2}} d\mathcal{H}_{(d)}^{\mathbb{R}}$$

donde $\tilde{A} \subseteq \mathcal{H}_{(d)}^{\mathbb{R}}$ es el cono sobre A y $\chi_{\tilde{A}} : \mathcal{H}_{(d)}^{\mathbb{R}} \rightarrow \{0, 1\}$ es la función característica de \tilde{A} .

A lo largo de este manuscrito haremos un uso de forma continuada de la *fórmula de la coárea*. La versión más general se debe a Federer (cf. [Fed69]) aunque algunas otras se pueden consultar en [How93]. Para nuestros propósitos nos podemos conformar con la versión *smooth* que se enuncia en [SS93b] (ver también [BCSS98]). Antes de enunciar el resultados necesitamos algunas definiciones previas:

Definición 1.3.1 (Jacobiano normal). *Sean X e Y dos variedades Riemanninas y sea $F : X \rightarrow Y$ una aplicación suprayectiva y C^1 . Sea $k := \dim(Y)$ la dimensión real de Y . Para todo punto regular $x \in X$ (i.e. F es una submersión en x), sea v_1^x, \dots, v_k^x una base ortonormal del subespacio vectorial $\ker(d_x F)^{\perp}$. Definimos el jacobiano normal de F en x , $\text{NJ}_x F$, como el volumen en $T_{F(x)}Y$ del paralelepípedo que resulta de expandir los vectores $d_x F(v_1^x), \dots, d_x F(v_k^x)$. En el caso de que $d_x F$ no sea suprayectiva, tomaremos $\text{NJ}_x F := 0$.*

A partir de esta definición, los siguiente resultado se siguen de forma natural:

Proposición 1.3.2. *Sean X e Y dos variedades Riemanninas, $F : X \rightarrow Y$ una aplicación C^1 y $x_1, x_2 \in X$ dos puntos. Asumiendo que existan dos isometrías, $\varphi_X : X \rightarrow X$ y $\varphi_Y : Y \rightarrow Y$ tales que*

$$\varphi_X(x_1) = x_2 \quad \text{y} \quad F \circ \varphi_X = \varphi_Y \circ F.$$

Entonces,

$$\text{NJ}_{x_1} F = \text{NJ}_{x_2} F.$$

Es más, si $G : Y \rightarrow X$ es la función inversa de F , entonces:

$$\text{NJ}_x F = \frac{1}{\text{NJ}_{F(x)} G}.$$

Teorema 1.3.3 (Fórmula de la coárea). *Sean X e Y dos variedades Riemanninas de dimensiones $k_1 \geq k_2$ respectivamente y sea $F : X \rightarrow Y$ una aplicación C^{∞} . Asumiendo que F sea una submersión en casi todo punto de X y que $\psi : X \rightarrow \mathbb{R}$ sea una función integrable entonces se cumple:*

$$\int_X \psi dX = \int_{y \in Y} \int_{x \in F^{-1}(y)} \psi(x) \frac{1}{\text{NJ}_x F} dF^{-1}(y) dY,$$

donde $\text{NJ}_x F$ denota el jacobiano normal de F en x .

El siguiente teorema es una adaptación al caso real de una proposición debida a Howard (cf. [How93]). Nótese que, en los siguientes resultados, el volumen de $\mathcal{O}(n+1, \mathbb{R})$ ha sido normalizado de forma que sea igual a 1.

Teorema 1.3.4. *Sea $V \subseteq \mathbb{P}_n(\mathbb{R})$ una variedad algebraica lisa de dimensión m . Entonces:*

$$\nu_m^{\mathbb{R}}[V] = \vartheta_m \int_{U \in \mathcal{O}(n+1, \mathbb{R})} \#(V \cap UL) d\mathcal{O}(n+1, \mathbb{R}),$$

donde $L \subseteq \mathbb{P}_n(\mathbb{R})$ es cualquier subespacio proyectivo de $\mathbb{P}_n(\mathbb{R})$ de codimensión m .

El siguiente corolario extiende al caso real un resultado clásico de Teoría de Eliminación.

Corolario 1.3.5. *Sea $f \in \mathcal{H}_{(d)}^{\mathbb{R}}$ con $f = (f_1, \dots, f_m)$ un sistema de polinomios homogéneos con coeficientes reales. Si asumimos que $(d) = (d_1, \dots, d_m)$ es un vector de números impares y que $V_{\mathbb{P}}(f) = \pi^{-1}(f) \subseteq \mathbb{P}_n(\mathbb{R})$ es una variedad algebraica lisa de codimensión m entonces*

$$\vartheta_{n-m} \leq \nu_{n-m}[V_{\mathbb{P}}(f)] \leq \mathcal{D}\vartheta_{n-m}.$$

Demostración. A lo largo de la demostración usaremos el Teorema 1.3.4. Sea $L \subseteq \mathbb{P}_n(\mathbb{R})$ un subespacio lineal de codimensión $n-m$. Entonces existe una matriz $A \in \mathcal{M}_{(n+1) \times m}(\mathbb{R})$ con $\text{rank}(M) = m$ tal que:

$$L = \left\{ (x_0, \dots, x_n) \in \mathbb{P}_n(\mathbb{R}) : A \begin{pmatrix} x_0 \\ \vdots \\ x_n \end{pmatrix} = 0 \right\}.$$

Para toda matriz ortogonal $U \in \mathcal{O}(n+1, \mathbb{R})$, tenemos:

$$UL = \left\{ (x_0, \dots, x_n) \in \mathbb{P}_n(\mathbb{R}) : AU^{-1} \begin{pmatrix} x_0 \\ \vdots \\ x_n \end{pmatrix} = 0 \right\}.$$

Para todo $f = (f_1, \dots, f_m) \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ existe un abierto no vacío de la topología de Zariski $R_f \subseteq \mathcal{O}(n+1, \mathbb{R})$ tal que para toda matriz $U \in R_f$:

- La variedad

$$W_U := \left\{ (x_0, \dots, x_n) \in \mathbb{P}_n(\mathbb{C}) : f_i(x) = 0, AU^{-1} \begin{pmatrix} x_0 \\ \vdots \\ x_n \end{pmatrix} = 0 \in \mathbb{C}^m \right\},$$

es una subvariedad proyectiva compleja de dimensión cero (y por lo tanto es un conjunto finito).

- El cardinal de W_U es el número de Bézout del sistema. Es decir:

$$\#W_U = d_1 \cdot \dots \cdot d_m \cdot 1 \cdot \dots \cdot 1 = \mathcal{D}.$$

Teniendo en cuenta que $W_U \cap \mathbb{P}_n(\mathbb{R}) = V_{\mathbb{P}}(f) \cap UL$ se sigue que

$$\#(V_{\mathbb{P}}(f) \cap UL) \leq \#W_U = \mathcal{D}, \quad \forall U \in R_f.$$

Por otro lado, como \mathcal{D} es impar, para todo $U \in R_f$, $W_U \cap \mathbb{P}_n(\mathbb{R}) \neq \emptyset$ y podemos concluir:

$$1 \leq \#(W_U \cap \mathbb{P}_n(\mathbb{R})) = \#(V_{\mathbb{P}}(f) \cap UL), \quad \forall U \in R_f.$$

Finalmente, como R_f es un abierto denso en la topología de Zariski sobre $\mathcal{O}(n+1, \mathbb{R})$, usando el Teorema 1.3.4 concluimos:

$$\begin{aligned} \vartheta_{n-m} &\leq \\ &\leq \vartheta_{n-m} \int_{R_f} \#(V_{\mathbb{P}}(f) \cap UL) d\mathcal{O}(n+1, \mathbb{R}) = \vartheta_{n-m} \int_{\mathcal{O}(n+1, \mathbb{R})} \#(V_{\mathbb{P}}(f) \cap UL) d\mathcal{O}(n+1, \mathbb{R}) \leq \\ &\leq \mathcal{D} \vartheta_{n-m} \end{aligned}$$

□

A lo largo de las siguientes páginas también haremos uso de los siguientes Teoremas clásicos de Probabilidad. Los enunciamos aquí por completitud.

Teorema 1.3.6 (Desigualdad de Chebysheff-Markov). *Sea (Ω, Σ, μ) un espacio de medida tal que $\nu[\Omega] < \infty$, $f : \Omega \rightarrow \mathbb{R}$ una función medible y $\varepsilon > 0$ entonces:*

$$\text{Prob}_{\Omega} [|f| \geq \varepsilon] \leq \frac{\mathbb{E}_{\Omega} [|f|]}{\varepsilon}$$

Teorema 1.3.7 (Desigualdad de Jensen). *Sea (Ω, Σ, μ) un espacio de medida tal que $\nu[\Omega] < \infty$ y X una variable aleatoria real. Entonces:*

$$\mathbb{E}_{\Omega} \left[\frac{1}{X} \right] \geq \frac{1}{\mathbb{E}_{\Omega}[X]}.$$

1.4 El número de condicionamiento

El impulsor del estudio del número de condicionamiento lineal en matrices fue A. Turing en su artículo [Tur48]. Posteriormente, fue desarrollado por von Neumann [vNG47] y Wilkinson [Wil65]. En los últimos años, autores como Smale [Sma85] y Edelman [Ede88, Ede92] han profundizado en su estudio, calculando las distribuciones de probabilidad del condicionamiento lineal tanto sobre el espacio de matrices complejas (ver [Ren94], [Dem88], [BP07a]) como sobre el espacio de matrices reales (véase [AW05], [BCL06], [CD05], [CAW03], [DM07]).

Un concepto importante a la hora de definir el condicionamiento lineal es la pseudo-inversa de Moore-Penrose.

Definición 1.4.1 (Pseudo-inversa de Moore-Penrose). *Sea $A \in \mathcal{M}_{m \times n}(\mathbb{K})$ una matriz sobre el cuerpo $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$. Llamaremos pseudo-inversa de Moore-Penrose a la única matriz A^\dagger tal que cumple simultáneamente las siguientes cuatro propiedades:*

$$1. AA^\dagger A = A \quad 2. A^\dagger AA^\dagger = A^\dagger \quad 3. (AA^\dagger)^* = AA^\dagger \quad 4. (A^\dagger A)^* = A^\dagger A$$

donde A^* denota la matriz transpuesta conjugada.

Una propiedad importante que posee la pseudo-inversa de Moore-Penrose es la siguiente. La demostración puede encontrarse en cualquier libro básico de Álgebra Lineal como por ejemplo [Gre81].

Lema 1.4.2. *Sea $A \in \mathcal{M}_{m \times n}(\mathbb{K})$ una matriz sobre el cuerpo $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ tal que el rango de A es máximo. Entonces se cumple:*

$$A^\dagger = \left(A|_{\ker(A)^\perp} \right)^{-1}.$$

En particular, si $D_z(f) : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ es suprayectiva, entonces se tiene que:

$$D_z(f)^\dagger = \left(D_z(f)|_{\ker(f)^\perp} \right)^{-1}.$$

Usaremos la siguiente versión del número de condicionamiento lineal para matrices cuadradas:

$$\kappa_D : \mathbb{P}(\mathcal{M}_n(\mathbb{R})) \rightarrow \mathbb{R}^+ \cup \{\infty\},$$

donde, para cualquier matriz cuadrada $A \in \mathbb{P}(\mathcal{M}_n(\mathbb{R}))$ definimos el condicionamiento lineal de A por:

$$\kappa_D(A) := \|A\|_F \|A^\dagger\|_2,$$

donde $\|A\|_F = (\text{Tr}(A^t A))^{1/2}$ es la norma de Frobenius, A^\dagger denota la pseudo-inversa de Moore-Penrose y $\|A^\dagger\|_2$ la norma de A^\dagger como operador lineal.

Nótese que $\kappa_D(A) = +\infty$ si y solo si $A : \mathbb{R}^n \rightarrow \mathbb{R}^m$ no es una aplicación lineal suprayectiva. Este número de condicionamiento lineal satisface el *Teorema del Número de Condicionamiento* de Eckart y Young (cf. [EY36] y también [BCSS98, BP07a]).

El siguiente resultado es una adaptación del resultado principal de [CD05].

Corolario 1.4.3. *Sea $m, n \in \mathbb{N}$ dos números positivos tales que $2 \leq m \leq n$ y sea $\varepsilon \leq \frac{1}{n-m+1}$ un número real positivo. Entonces*

$$\frac{1}{\sqrt{2\pi}} \left(\frac{nc\varepsilon}{n-m+1} \right)^{n-m+1} \leq \text{Prob}_{\mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R}))} [\kappa_D \geq \varepsilon^{-1}] \leq \frac{1}{\sqrt{2\pi}} \left(\frac{n^{3/2}C\varepsilon}{n-m+1} \right)^{n-m+1}.$$

donde $0.245 \leq c \leq 2$, $5.013 \leq C \leq 6.414$ y $\text{Prob}_{\mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R}))}$ denota la distribución de probabilidad asociada a la estructura Riemannian canónica en $\mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R})) = \mathbb{P}_{nm-1}(\mathbb{R})$.

Demostración. Sean A_ε , B_ε y C_ε los siguientes conjuntos:

$$\begin{aligned} B_\varepsilon &= \{M \in \mathcal{M}_{m \times n}(\mathbb{R}) : \kappa_D(M) > \varepsilon^{-1}\}, \\ C_\varepsilon &= \{M \in \mathcal{M}_{m \times n}(\mathbb{R}) : \kappa_D(M) > \sqrt{n}\varepsilon^{-1}\}, \\ A_\varepsilon &= \{M \in \mathcal{M}_{m \times n}(\mathbb{R}) : \kappa_2(M) > \varepsilon^{-1}\}, \end{aligned}$$

donde κ_2 denota el número de condicionamiento tradicional. Es decir:

$$\kappa_2(M) = \|A\|_2 \|A^\dagger\|_2.$$

A partir de ambas definiciones es claro que: para toda matriz M ,

$$\frac{\kappa_D(M)}{\sqrt{n}} \leq \kappa_2(M) \leq \kappa_D(M).$$

Inmediatamente concluimos que:

$$C_\varepsilon \subseteq A_\varepsilon \subseteq B_\varepsilon,$$

y por lo tanto:

$$\mathbb{P}_{\mathcal{M}_{m \times n}(\mathbb{R})}(C_\varepsilon) \subseteq \mathbb{P}_{\mathcal{M}_{m \times n}(\mathbb{R})}(A_\varepsilon) \subseteq \mathbb{P}_{\mathcal{M}_{m \times n}(\mathbb{R})}(B_\varepsilon).$$

Usando las cotas del teorema principal de [CD05] y ajustando las constantes obtenemos:

$$\frac{1}{\sqrt{2\pi}} \left(\frac{n c \varepsilon}{n - m + 1} \right)^{n-m+1} \leq \text{Prob}_{\mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R}))} [\kappa_D \geq \varepsilon^{-1}] \leq \frac{1}{\sqrt{2\pi}} \left(\frac{n^{3/2} C \varepsilon}{n - m + 1} \right)^{n-m+1}.$$

□

Resumimos ahora algunos aspectos relacionados con el número de condicionamiento de sistema de ecuaciones polinomiales homogéneas multivariadas introducido en el artículo [SS93a] (cf. también [BCSS98] y [Ded06]).

Definición 1.4.4. Sea $(f, \zeta) \in V^{\mathbb{R}}$ un punto en variedad de incidencia. Definimos el número de condicionamiento normalizado (en adelante condicionamiento μ_{norm}) de f en ζ como:

$$\mu_{norm}^m(f, \zeta) := \|f\|_\Delta \|\|\zeta\|_2 (T_\zeta f)^\dagger \Delta ((d)^{1/2})\|_2,$$

donde $\|\cdot\|_\Delta$ denota la norma inducida por el producto escalar de Bombieri, $\|\cdot\|$ es la norma como operador lineal, $(T_x f)^\dagger$ denota la pseudo-inversa de Moore-Penrose de $T_x f : T_x \mathbb{P}_n(\mathbb{R}) \rightarrow T_0 \mathbb{R}^n = \mathbb{R}^n$ y $\Delta((d)^{1/2}) \in \mathcal{M}_m(\mathbb{R})$ es la matriz real

$$\Delta((d)^{1/2}) := \begin{pmatrix} d_1^{1/2} & & & \\ & \ddots & & \\ & & & d_m^{1/2} \end{pmatrix}.$$

Nótese que en el caso que tomemos representantes de f y ζ de norma 1 la formula queda simplificada a:

$$\mu_{norm}^m(f, \zeta) := \|(T_\zeta f)^\dagger \Delta((d)^{1/2})\|_2 = \frac{\kappa_D((T_\zeta f)^\dagger \Delta((d)^{1/2}))}{\|(T_\zeta f)^\dagger \Delta((d)^{1/2})\|_F},$$

donde $\|\cdot\|_F$ denota la norma de Frobenius y $\kappa_D(\cdot)$ el condicionamiento lineal.

Nótese que en el caso que $m = n$, el condicionamiento $\mu_{norm}^n(f, \zeta)$ equivale al número de condicionamiento habitual $\mu_{norm}(f, \zeta)$ para sistemas de ecuaciones definido en [SS93a].

Es de notar también que $\mu_{norm}^m(f, \zeta) = +\infty$ si y sólo si $\zeta \in V_{\mathbb{P}}^{\mathbb{R}}(f)$ es un punto singular de la aplicación $f : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^m$.

El condicionamiento μ_{norm} es invariante bajo la acción del grupo ortogonal $\mathcal{O}(m+1, \mathbb{R})$ sobre la variedad de incidencia V . Es decir, para toda matriz ortogonal $U \in \mathcal{O}(m+1, \mathbb{R})$ y cada punto $(f, \zeta) \in V^{\mathbb{R}}$ en la variedad de incidencia

$$\mu_{norm}^m(f, \zeta) = \mu_{norm}^m(f \circ U^{-1}, U\zeta).$$

El siguiente teorema se sigue esencialmente de un enunciado similar en [SS93b] (cf. también [BP07b]).

Teorema 1.4.5. *Para cualquier par $(f, \zeta) \in V^{\mathbb{R}}$ la siguiente igualdad se cumple:*

$$\mu_{norm}^m(f, \zeta) = \frac{1}{d_{\mathbb{P}}((f, \zeta), \Sigma^{\mathbb{R}} \cap V_{\zeta})},$$

donde $d_{\mathbb{P}}(\cdot, \cdot)$ es la distancia proyectiva inducida en $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ por el producto escalar de Bombieri.

Tal y como se muestra en [SS93b] podemos descomponer $V_{e_0}^{\mathbb{R}}$ como la suma ortogonal de dos subespacios vectoriales

$$V_{e_0}^{\mathbb{R}} = V'_{e_0} \perp L_{e_0},$$

donde L_{e_0} es el complemento ortogonal de V'_{e_0} y este a su vez es el subespacio vectorial de $V_{e_0}^{\mathbb{R}}$ dado por las siguientes igualdades:

$$V'_{e_0} := \{f \in V_{e_0}^{\mathbb{R}} : T_{e_0}f = D_{e_0}f|_{\langle e_0 \rangle^{\perp}} \equiv 0\}.$$

Nótese que V'_{e_0} es el subespacio vectorial de todos los sistemas $f = (f_1, \dots, f_m)$ tales que la multiplicidad de e_0 en f_i es al menos 2.

Otra descripción de los elementos de L_{e_0} es: dado un sistema $f = (f_1, \dots, f_m)$, f pertenece a L_{e_0} si y solo si existen aplicaciones lineales $l_1, \dots, l_m : \mathbb{R}^n \rightarrow \mathbb{R}$ tales que para todo número natural $i, 1 \leq i \leq m$ se cumple:

$$f_i = x_0^{d_i-1} l_i.$$

A partir de esta descripción se sigue que podemos identificar el conjunto de matrices $\mathcal{M}_{m \times n}(\mathbb{R})$ con L_{e_0} mediante la siguiente aplicación: A toda matriz $M \in \mathcal{M}_{m \times n}(\mathbb{R})$ le asociamos el sistema $\varphi(M) := (f_1, \dots, f_m) \in L_{e_0}$ dado por la siguiente igualdad:

$$\begin{pmatrix} f_1 \\ \vdots \\ f_m \end{pmatrix} := \begin{pmatrix} x_0^{d_1-1} & & \\ & \ddots & \\ & & x_0^{d_m-1} \end{pmatrix} M \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}.$$

Nótese que la aplicación $\varphi : \mathcal{M}_{m \times n}(\mathbb{R}) \rightarrow L_{e_0}$ es una isometría cuando hemos dotado a $\mathcal{M}_{m \times n}(\mathbb{R})$ con la norma de Frobenius y a L_{e_0} con la norma inducida por el producto escalar de Bombieri $\langle \cdot, \cdot \rangle_{\Delta}$ definido sobre $\mathcal{H}_{(d)}^{\mathbb{R}}$.

Usando la identificación anterior podemos crear una relación entre el condicionamiento de un sistema de ecuaciones y el condicionamiento lineal.

Proposición 1.4.6. *Usando la notación anterior. La siguiente aplicación es una isometría*

$$\begin{aligned} \rho : B^1(L_{e_0}) &\longrightarrow B^1(\mathcal{M}_{m \times n}(\mathbb{R})) \\ f &\longmapsto \Delta((d)^{-1/2}) T_{e_0} f, \end{aligned}$$

donde $B^1(L_{e_0})$ es la bola cerrada de radio 1 en L_{e_0} respecto de la norma inducida por el producto escalar de Bombieri $\langle \cdot, \cdot \rangle_\Delta$ en L_{e_0} y $B^1(\mathcal{M}_{m \times n}(\mathbb{R}))$ es la bola cerrada de radio 1 en $\mathcal{M}_{m \times n}(\mathbb{R})$ respecto de la norma de Frobenius. Además, se tiene que para todo $f \in L_{e_0}$,

$$\mu_{norm}(f, e_0) = \kappa_\Delta(\rho(f)).$$

Demostración. La primera parte se sigue simplemente a partir de las definiciones de la norma de Frobenius y norma.

Para la segunda parte se sigue inmediatamente de las definiciones del condicionamiento μ_{norm} y condicionamiento lineal teniendo en cuenta que:

$$\|f\|_\Delta = \|\Delta((d)^{-1/2}) T_{e_0} f\|_F \quad \text{y} \quad \|e_0\|_2 = 1.$$

□

Capítulo 2

Algunos resultados integrales básicos

God does not care about our mathematical difficulties. He integrates empirically¹.
Albert Einstein [Inf42, pág. 222]

En este capítulo incluiremos algunos resultados básicos relacionados con la variedad de incidencia que usaremos a lo largo de esta memoria.

Sean $x \in \mathbb{P}_n(\mathbb{R})$ y $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ dos elementos y sean \tilde{x} y \tilde{f} representantes afines respectivos de x y f tales que $\|\tilde{x}\|_2 = \|\tilde{f}\|_{\Delta} = 1$. Consideremos los espacios ortogonales x^{\perp} y f^{\perp} . En la introducción, hemos definido las aplicaciones $\varphi_{\tilde{x}}$ y $\varphi_{\tilde{f}}$ como sigue:

$$\begin{aligned} \varphi_{\tilde{x}}: x^{\perp} &\longrightarrow \mathbb{P}_n(\mathbb{R}) \setminus x^{\perp}, \\ y &\mapsto \tilde{x} + y \\ \varphi_{\tilde{f}}: f^{\perp} &\longrightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus f^{\perp}, \\ g &\mapsto \tilde{f} + g, \end{aligned}$$

de las cuales sabemos que son difeomorfismos y además isometría en 0 (esto es, $d_0\varphi_{\tilde{x}}$ y $d_0\varphi_{\tilde{f}}$ son isometría). El siguiente resultado (véase [BCSS98, pág. 193]) resume algunas de las propiedades de la variedad de incidencia. Incluimos la demostración por completitud de esta memoria, dado que se trata de un resultado esencial en nuestro esquema.

Proposición 2.0.7. *La variedad de incidencia $V^{\mathbb{R}}$ es una variedad diferenciable de dimensión real $N + n - m$. Además, sea $(f, \zeta) \in V^{\mathbb{R}}$ un punto cualquiera y $(\tilde{f}, \tilde{\zeta})$ representantes afines respectivos del par (f, ζ) tales que $\|\tilde{f}\|_{\Delta} = \|\tilde{\zeta}\|_2 = 1$. Entonces, el espacio tangente $T_{(f, \zeta)}V^{\mathbb{R}} \subseteq f^{\perp} \times \zeta^{\perp}$ puede identificarse con el espacio vectorial definido a continuación:*

$$T_{(f, \zeta)}V^{\mathbb{R}} \equiv \{(g, x) \in f^{\perp} \times \zeta^{\perp} : g(\tilde{z}) + (d_{\tilde{z}}\tilde{f})x = 0\}.$$

La identificación viene dada por la isometría

$$d_{(0,0)}(\varphi_{\tilde{f}} \times \varphi_{\tilde{\zeta}}).$$

¹A Dios no le importan las dificultades matemáticas. Él integra de forma empírica.

2. Algunos resultados integrales básicos

Demostración. Sea $(f, \zeta) \in V^{\mathbb{R}}$ un punto, y sean $\tilde{f}, \tilde{\zeta}$ representantes afines respectivos de f, ζ , tales que $\|\tilde{f}\|_{\Delta} = \|\tilde{\zeta}\|_2 = 1$. Entonces, la siguiente aplicación es un difeomorfismo sobre su imagen:

$$\begin{aligned} \varphi_{\tilde{f}} \times \varphi_{\tilde{\zeta}} : f^{\perp} \times \zeta^{\perp} &\longrightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \times \mathbb{P}_n(\mathbb{R}) \\ (g, x) &\mapsto (\tilde{f} + g, \tilde{\zeta} + x). \end{aligned}$$

Sea $\widetilde{V}^{\mathbb{R}} := (\varphi_{\tilde{f}} \times \varphi_{\tilde{\zeta}})^{-1}(V^{\mathbb{R}})$ la anti-imagen de $V^{\mathbb{R}}$ mediante este difeomorfismo. Es decir,

$$\widetilde{V}^{\mathbb{R}} := \{(g, x) \in f^{\perp} \times \zeta^{\perp} : (\tilde{f} + g)(\tilde{\zeta} + x) = 0\}.$$

Entonces, $\widetilde{V}^{\mathbb{R}}$ también es la anti-imagen de 0 mediante la aplicación diferenciable

$$\begin{aligned} \varphi : f^{\perp} \times \zeta^{\perp} &\longrightarrow \mathbb{R}^n \\ (g, x) &\mapsto (\tilde{f} + g)(\tilde{\zeta} + x). \end{aligned}$$

Ahora, se comprueba trivialmente que 0 es un valor regular de esta aplicación. Esto es, para todo punto $(g, x) \in \widetilde{V}^{\mathbb{R}}$, $d_{(g,x)}\varphi$ es suprayectiva. Por lo tanto, $\widetilde{V}^{\mathbb{R}}$ es una variedad diferenciable real de dimensión $N + n - m$. Deducimos que $V^{\mathbb{R}}$ es también una variedad diferenciable real de dimensión $N + n - m$.

Además, un vector $(g, x) \in f^{\perp} \times \zeta^{\perp}$ pertenece a $T_{(0,0)}\widetilde{V}^{\mathbb{R}}$ si y sólo si

$$g(\tilde{\zeta}) + d_{\tilde{\zeta}}\tilde{f}(x) = 0,$$

lo que termina la demostración. □

Como hemos indicado en la introducción, para cada matriz ortogonal $U \in \mathcal{O}(n + 1, \mathbb{R})$, se definen isometrías en $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ y $\mathbb{P}_n(\mathbb{R})$, como sigue:

$$\begin{aligned} \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) &\longrightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}), \\ f &\mapsto f \circ U \\ \mathbb{P}_n(\mathbb{R}) &\longrightarrow \mathbb{P}_n(\mathbb{R}) \\ x &\mapsto U^{-1}x. \end{aligned}$$

A partir de estas dos isometrías queda definida otra isometría en $V^{\mathbb{R}}$:

$$\begin{aligned} V^{\mathbb{R}} &\longrightarrow V^{\mathbb{R}}, \\ (f, \zeta) &\mapsto (f \circ U, U^{-1}\zeta) \end{aligned}$$

El siguiente teorema generaliza ampliamente un resultado de Shub y Smale (véase por ejemplo [BCSS98, Prop. 2, pág. 244] o [SS93a]). Aunque el esquema de la prueba es similar la mostramos a continuación por completitud.

Teorema 2.0.8. *Sea $\phi : W \longrightarrow \mathbb{R}$ una función integrable, tal que para toda matriz ortogonal $U \in \mathcal{O}(n + 1, \mathbb{R})$, se satisface que:*

$$\phi(f, \zeta) = \phi(f \circ U, U^{-1}\zeta).$$

Sea \mathcal{J} la integral definida a continuación:

$$\mathcal{J} := \int_{(f,\zeta) \in V^{\mathbb{R}}} \phi(f, \zeta) \text{NJ}_{(f,\zeta)} \pi_1 dV^{\mathbb{R}}.$$

Entonces, las siguientes igualdades satisfacen:

$$\mathcal{J} = \int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \int_{\zeta \in V^{\mathbb{R}}(f)} \phi(f, \zeta) dV^{\mathbb{R}}(f) d\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) = \vartheta_n \int_{f \in V_{e_0}^{\mathbb{R}}} \phi(f, e_0) \frac{\text{NJ}_{(f,e_0)} \pi_1}{\text{NJ}_{(f,e_0)} \pi_2} dV_{e_0}^{\mathbb{R}}.$$

Demostración. La primera de las dos igualdades resulta del Teorema 1.3.3 aplicado a π_1 . Para la segunda, también el Teorema 1.3.3 nos asegura que:

$$\mathcal{J} = \int_{x \in \mathbb{P}_n(\mathbb{R})} \int_{f \in V_x^{\mathbb{R}}} \phi(f, x) \frac{\text{NJ}_{(f,x)} \pi_1}{\text{NJ}_{(f,x)} \pi_2} dV_x^{\mathbb{R}} d\mathbb{P}_n(\mathbb{R}).$$

Ahora, sea $x \in \mathbb{P}_n(\mathbb{R})$ un punto proyectivo cualquiera y sea $U \in \mathcal{O}(n+1, \mathbb{R})$ una matriz ortogonal tal que $Ue_0 = x$. Entonces, la aplicación que manda f a $f \circ U$ es una isometría de $V_x^{\mathbb{R}}$ en $V_{e_0}^{\mathbb{R}}$. Por lo tanto,

$$\int_{f \in V_x^{\mathbb{R}}} \phi(f, x) \frac{\text{NJ}_{(f,x)} \pi_1}{\text{NJ}_{(f,x)} \pi_2} dV_x^{\mathbb{R}} = \int_{f \in V_{e_0}^{\mathbb{R}}} \phi(f \circ U^{-1}, Ue_0) \frac{\text{NJ}_{(f \circ U^{-1}, Ue_0)} \pi_1}{\text{NJ}_{(f \circ U^{-1}, Ue_0)} \pi_2} dV_{e_0}^{\mathbb{R}}.$$

Como $\phi(f \circ U^{-1}, Ue_0) = \phi(f, x)$. Además, las aplicaciones definidas a continuación son isometrías:

$$\begin{array}{ccc} V^{\mathbb{R}} & \longrightarrow & V^{\mathbb{R}}, & \mathbb{P}_n(\mathbb{R}) & \longrightarrow & \mathbb{P}_n(\mathbb{R}) \\ (g, x) & \mapsto & (g \circ U^{-1}, Ux) & x & \mapsto & Ux \end{array}$$

Por tanto, estamos en las condiciones del Corolario 1.3.2. Deducimos que

$$\text{NJ}_{(f \circ U^{-1}, Ue_0)} \pi_2 = \text{NJ}_{(f, e_0)} \pi_2.$$

Un argumento simétrico con la aplicación $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \longrightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ dada como $f \mapsto f \circ U^{-1}$ demuestra que

$$\text{NJ}_{(f \circ U^{-1}, Ue_0)} \pi_1 = \text{NJ}_{(f, e_0)} \pi_1,$$

y el teorema queda demostrado. \square

A continuación calculamos el valor de los jacobianos normales que aparecen en el Teorema 2.0.8. De nuevo, el resultado que sigue generaliza los cálculos realizados por Shub y Smale en [SS93a] (véase también Lemas 2 y 3 de [BCSS98, págs. 241–243]).

Lema 2.0.9. *Sea $f \in V_{e_0}^{\mathbb{R}}$ un sistema tal que $\text{rank}(T_{e_0} f) = m$. Entonces, se tiene:*

$$\begin{aligned} \text{NJ}_{(f, e_0)} \pi_1 &= \frac{1}{\det((Id_m + (T_{e_0} f^\dagger)^t (T_{e_0} f^\dagger)))^{1/2}}, \\ \text{NJ}_{(f, e_0)} \pi_2 &= \frac{1}{\det((Id_m + (T_{e_0} f)(T_{e_0} f)^t))^{1/2}}. \end{aligned}$$

2. Algunos resultados integrales básicos

Demostración. Como hemos visto en la Proposición 2.0.7,

$$T_{(f,e_0)}V^{\mathbb{R}} \equiv \{(g, x) \in f^{\perp} \times e_0^{\perp} : g(e_0) + (T_{e_0}f)x = 0\},$$

donde hemos elegido un representante de norma 1 de f . Sea K_1 el núcleo de la matriz diferencial de π_1 en (f, e_0) . Esto es, $K_1 := \ker(d_{(f,e_0)}\pi_1)$. Entonces, $K_1 = \{(0, x) : x \in \ker(T_{e_0}f)\}$ y

$$\text{NJ}_{(f,e_0)}\pi_1 = \text{NJ}_{(0,0)}(d_{(f,e_0)}\pi_1)|_{K_1^{\perp}} = \frac{1}{\text{NJ}_{(0,0)}((d_{(f,e_0)}\pi_1)|_{K_1^{\perp}})^{-1}} = \frac{1}{\text{NJ}_{(0,0)}(d_{(f,e_0)}\pi_1)^{\dagger}}.$$

Sea β una base ortonormal de f^{\perp} tal que los primeros m elementos de la base son los sistemas

$$\begin{aligned} \beta_1 &:= [X_0^{d_1}, 0, \dots, 0], \\ &\vdots \\ \beta_m &:= [0, \dots, 0, X_0^{d_m}]. \end{aligned}$$

Observamos que las primeras m coordenadas de cualquier sistema $g := [g_1, \dots, g_m] \in \mathcal{H}_{(d)}^{\mathbb{R}}$ en esta bases coinciden exactamente con

$$g(e_0) = (g_1(e_0), \dots, g_m(e_0)).$$

Además, tenemos las siguientes propiedades:

- $(d_{(f,e_0)}\pi_1)^{\dagger}(\beta_i) = (\beta_i, x_i)$ con $x_i := -(T_{e_0}f)^{\dagger}(e_i)$, para $1 \leq i \leq m$.
- $(d_{(f,e_0)}\pi_1)^{\dagger}(v) = (v, 0)$, para $v \in \beta, v \notin \{\beta_1, \dots, \beta_m\}$.

Por lo tanto,

$$\text{NJ}_{(0,0)}(d_{(f,e_0)}\pi_1)^{\dagger} = \det(Id_m + ((T_{e_0}f)^{\dagger})^t (T_{e_0}f)^{\dagger}).$$

En cuanto a π_2 , obsérvese de nuevo que

$$\text{NJ}_{(f,e_0)}\pi_2 = \frac{1}{\text{NJ}_{(0,0)}(d_{(f,e_0)}\pi_2)^{\dagger}}.$$

Obtenemos la siguiente igualdad

$$\ker(d_{(f,e_0)}\pi_2)^{\perp} = \{(g, 0) : g(e_0) = 0\}^{\perp} = \langle \beta_1, \dots, \beta_m \rangle \times e_0^{\perp},$$

donde $\langle \beta_1, \dots, \beta_m \rangle$ denota simplemente el subespacio generado por esos vectores. Por lo tanto,

$$(d_{(f,e_0)}\pi_2)^{\dagger}(e_i) = (g_i, e_i), \quad 1 \leq i \leq n.$$

donde las primeras m coordenadas de g_i en la base β forman el vector $-(T_{e_0}f)e_i$, y el resto de coordenadas son nulas. Concluimos que

$$\text{NJ}_{(0,0)}(d_{(f,e_0)}\pi_2)^{\dagger} = \det(Id_n + (T_{e_0}f)^t (T_{e_0}f))^{1/2}.$$

Finalmente, sea $T_{e_0}f = U(D \ 0)V^t$ una descomposición en valores singulares de $T_{e_0}f$. Entonces, tenemos

$$\det (Id_n + (T_{e_0}f)^t(T_{e_0}f)) = \det \left(Id_n + V \begin{pmatrix} D^2 & 0 \\ 0 & 0 \end{pmatrix} V^t \right) = \det (Id_m + D^2),$$

y además

$$\det (Id_m + (T_{e_0}f)(T_{e_0}f)^t) = \det (Id_m + UD^2U^t) = \det (Id_m + D^2),$$

luego deducimos que

$$\det (Id_n + (T_{e_0}f)^t(T_{e_0}f)) = \det (Id_m + (T_{e_0}f)(T_{e_0}f)^t),$$

lo que termina la demostración del lema. □

Inmediatamente se sigue el siguiente resultado, que en el caso $n = m$ fue formulado por Shub y Smale en [SS93a] (véase también el Lema 3 de [BCSS98, págs. 241–243]). Desarrollamos aquí el caso general.

Lema 2.0.10. *Usando la notación anterior. Sea $f \in V_{e_0}^{\mathbb{R}}$ un sistema tal que $\text{rank}(T_{e_0}f) = m$ entonces:*

$$\frac{\text{NJ}_{(f,e_0)}\pi_1}{\text{NJ}_{(f,e_0)}\pi_2} = \det ((T_{e_0}f)(T_{e_0}f)^t)^{1/2}.$$

Demostración. Por el Lema 2.0.9,

$$\frac{\text{NJ}_{(f,e_0)}\pi_1}{\text{NJ}_{(f,e_0)}\pi_2} = \frac{\det (Id_m + BB^t)^{1/2}}{\det (Id_m + (B^\dagger)^t(B^\dagger))^{1/2}},$$

donde $B := T_{e_0}f \in \mathcal{M}_{m \times n}(\mathbb{R})$. Entonces,

$$\frac{1}{\det (BB^t)^{1/2}} \frac{\text{NJ}_{(f,e_0)}\pi_1}{\text{NJ}_{(f,e_0)}\pi_2} = \frac{\det (Id_m + BB^t)^{1/2}}{\det (BB^t + BB^t(B^\dagger)^t B^\dagger)^{1/2}}.$$

Ahora, observamos que $BB^t(B^\dagger)^t B^\dagger = B(B^\dagger B)^t B^\dagger$. Además, por las propiedades elementales de la pseudoinversa de Moore-Penrose, $B^\dagger B \in \mathcal{M}_n(\mathbb{R})$ es autoadjunta. Esto es,

$$B^\dagger B = (B^\dagger B)^t.$$

Por último, B es suprayectiva, luego $BB^\dagger = Id_m$. Por lo tanto,

$$\det (BB^t + BB^t(B^\dagger)^t B^\dagger) = \det (BB^t + BB^\dagger BB^t) = \det (BB^t + Id_m),$$

y el lema queda demostrado. □

Continuamos con varias fórmulas que nos permitirán realizar las estimaciones en las siguientes secciones.

2. Algunos resultados integrales básicos

Proposición 2.0.11. *Sea $\phi : V^{\mathbb{R}} \longrightarrow \mathbb{R}$ una función positiva e integrable. Si además ϕ es invariante respecto a la acción del grupo ortogonal $\mathcal{O}(n+1, \mathbb{R})$ sobre $V^{\mathbb{R}}$ se tiene:*

$$\int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \int_{\zeta \in V_{\mathbb{P}}^{\mathbb{R}}(f)} \phi(f, \zeta) dV_{\mathbb{P}}^{\mathbb{R}}(f) d\nu_{\Delta} = \vartheta_n \int_{f \in V_{e_0}^{\mathbb{R}}} \phi(f, e_0) |\det((T_{e_0}f)(T_{e_0}f)^t)|^{1/2} dV_{e_0}^{\mathbb{R}}$$

Demostración. Inmediata a partir de las proposiciones 2.0.7 y 2.0.10. \square

Proposición 2.0.12. *Usando la notación anterior. Sea $\phi : V^{\mathbb{R}} \longrightarrow \mathbb{R}$ una función integrable tal que para todo $f \in V_{e_0}^{\mathbb{R}}$ que cumple*

$$\phi(f, e_0) = \phi(T_{e_0}f, e_0) = \phi(g, e_0),$$

con $g \in L_{e_0}$ la componente de f en la descomposición ortogonal $V_{e_0}^{\mathbb{R}} = V'_{e_0} \perp L_{e_0}$. Entonces

$$\int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \int_{\zeta \in V_{\mathbb{P}}^{\mathbb{R}}(f)} \phi(f, z) dV_{\mathbb{P}}^{\mathbb{R}}(f) d\nu_{\Delta} = 2\vartheta_n \vartheta_{N-(n+1)m-1} I(\phi, m, n, (d)),$$

donde $N = \dim(\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}))$ y

$$I(\phi, m, n, (d)) = \int_{B^1(L_{e_0})} \phi(f, e_0) |\det((T_{e_0}f)(T_{e_0}f)^t)|^{1/2} (1 - \|f\|_{\Delta}^2)^{\frac{N-(n+1)m-2}{2}} dB^1,$$

con $B^1(L_{e_0})$ es la bola cerrada de radio 1 en L_{e_0} respecto de la norma inducida por el producto escalar de Bombieri $\langle \cdot, \cdot \rangle_{\Delta}$ en L_{e_0} .

Demostración. Sea $\mathbb{S}(V_{e_0}^{\mathbb{R}})$ la esfera unitaria respecto de la norma inducida por el producto escalar de Bombieri en \tilde{V}_{e_0} . Denotamos por $p : \mathbb{S}(V_{e_0}^{\mathbb{R}}) \longrightarrow B^1(L_{e_0})$ la proyección canónica sobre la bola unitaria respecto de la norma inducida por el producto escalar de Bombieri en L_{e_0} . Siguiendo el Ejemplo 9 y el Lema 2 del Capítulo 11 de [BCSS98] tenemos:

- Para todo punto $g \in B^1(L_{e_0})$, la fibra $p^{-1}(g)$ puede identificarse con una esfera en $V_{e_0}^{\mathbb{R}}$ de dimensión $(\dim(V_{e_0}^{\mathbb{R}}) - \dim(L_{e_0}) - 1)$ y radio $(1 - \|g\|_{\Delta}^2)^{1/2}$. Es decir:

$$p^{-1}(g) \equiv \text{esfera de dimensión } (N - (n+1)m - 1) \text{ y radio } (1 - \|g\|_{\Delta}^2)^{1/2} \text{ en } V_{e_0}^{\mathbb{R}}.$$

- El jacobiano normal de p en cualquier $f \in \mathbb{S}(V_{e_0}^{\mathbb{R}})$ satisface:

$$\text{NJ}_f p = (1 - \|p(f)\|_{\Delta}^2)^{1/2}.$$

Se sigue inmediatamente que

$$\begin{aligned} & \int_{f \in \mathbb{S}(V_{e_0}^{\mathbb{R}})} \phi(f, e_0) |\det((T_{e_0}f)(T_{e_0}f)^t)|^{1/2} d\mathbb{S} = \\ &= \int_{g \in B^1(L_{e_0})} \int_{x \in p^{-1}(g)} \phi(g, e_0) |\det((T_{e_0}g)(T_{e_0}g)^t)|^{1/2} (1 - \|g\|_{\Delta}^2)^{-1/2} dp^{-1}(g) dB^1 = \\ &= \int_{g \in B^1(L_{e_0})} \phi(g, e_0) |\det((T_{e_0}g)(T_{e_0}g)^t)|^{1/2} (1 - \|g\|_{\Delta}^2)^{-1/2} \nu^{\mathbb{R}}[p^{-1}(g)] dB^1 \end{aligned}$$

Ahora, como $p^{-1}(g)$ es una esfera real de dimensión $(N - (n + 1)m - 1)$ y radio $(1 - \|g\|_\Delta^2)^{1/2}$ tenemos

$$\nu^\mathbb{R}[p^{-1}(g)] = 2(1 - \|g\|_\Delta^2)^{\frac{N-(n+1)m-1}{2}} \vartheta_{N-(n+1)m-1}.$$

Concluimos entonces:

$$\int_{f \in V_{e_0}^\mathbb{R}} \phi(f, e_0) |\det((T_{e_0}f)(T_{e_0}f)^t)|^{1/2} dV_{e_0}^\mathbb{R} = 2\vartheta_{N-(n+1)m-1} I(\phi, m, n, (d)),$$

donde

$$I(\phi, m, n, (d)) = \int_{g \in B^1(L_{e_0})} \phi(g, e_0) |\det((T_{e_0}g)(T_{e_0}g)^t)|^{1/2} (1 - \|g\|_\Delta^2)^{\frac{N-(n+1)m-2}{2}} dB^1$$

□

Proposición 2.0.13. Usando la notación anterior. Sea $\phi : V^\mathbb{R} \rightarrow \mathbb{R}$ una función integrable tal que para todo $f \in V_{e_0}^\mathbb{R}$:

$$\phi(f, e_0) = \frac{\phi(g, e_0)}{\|g\|_\Delta},$$

con $g \in L_{e_0}$ la componente de f en la descomposición ortogonal $V_{e_0}^\mathbb{R} = V'_{e_0} \perp L_{e_0}$. Entonces

$$\int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R})} \int_{z \in V_{\mathbb{P}}^\mathbb{R}(f)} \phi(f, z) dV_{\mathbb{P}}^\mathbb{R}(f) d\nu_\Delta = 2\vartheta_n \vartheta_{N-(n+1)m-1} J(\phi, m, n, (d)),$$

donde $N = \dim(\mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R}))$ y

$$J(\phi, m, n, (d)) = \int_{g \in B^1(L_{e_0})} \frac{\phi(g, e_0)}{\|g\|_\Delta} |\det((T_{e_0}f)(T_{e_0}f)^t)|^{1/2} (1 - \|f\|_\Delta^2)^{\frac{N-(n+1)m-2}{2}} dB^1,$$

con $B^1(L_{e_0})$ es la bola cerrada de radio 1 en L_{e_0} respecto de la norma inducida por el producto escalar de Bombieri $\langle \cdot, \cdot \rangle_\Delta$ en L_{e_0} .

Demostración. La prueba es esencialmente la misma que la anterior proposición y por ello la omitimos. □

Por último mostramos otra identidad muy útil.

Proposición 2.0.14. Usando la notación anterior.

$$\int_{(\mathcal{R}_{(d)})_{e_0}} \frac{1}{\nu_{n-m}[V_{\mathbb{P}}(f)]} \frac{\text{NJ}_{(f,e_0)}\pi_1}{\text{NJ}_{(f,e_0)}\pi_2} dV_{e_0} = \frac{\nu_\Delta[\mathcal{R}_{(d)}]}{\vartheta_n},$$

donde $\mathcal{R}_{(d)} := \pi_1(V) = \{f \in \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R}) : V_{\mathbb{P}}(f) \neq \emptyset\}$ y $(\mathcal{R}_{(d)})_{e_0} = \pi_1(V_{e_0})$.

2. Algunos resultados integrales básicos

Demostración. Para empezar, se tiene que

$$I := \int_{\mathcal{R}_{(d)}} \frac{1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}(f)]} \left(\int_{\mathbb{V}_{\mathbb{P}}(f)} dV_{\mathbb{P}}(f) \right) d\nu_{\Delta} = \int_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \chi_{\mathcal{R}_{(d)}} d\nu_{\Delta} = \nu_{\Delta}[\mathcal{R}_{(d)}].$$

Por otro lado, teniendo en cuenta que $\mathbb{V}_{\mathbb{P}}(f) = \pi_1^{-1}(f)$ y aplicando la fórmula de la coárea obtenemos

$$I = \int_{(f,x) \in V^{\mathbb{R}}} \frac{\text{NJ}_{(f,x)}\pi_1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}(f)]} dV = \int_{x \in \mathbb{P}_n(\mathbb{R})} \int_{f \in \pi_2^{-1}(x)} \frac{1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}(f)]} \frac{\text{NJ}_{(f,x)}\pi_1}{\text{NJ}_{(f,x)}\pi_2} d\pi_2^{-1}(x) d\nu_{\mathbb{P}}.$$

Como $\nu_{n-m}[\mathbb{V}_{\mathbb{P}}(f)]$ es invariante bajo la acción del grupo ortogonal $\mathcal{O}(n+1, \mathbb{R})$, concluimos

$$I = \vartheta_n \int_{f \in V_{e_0}^{\mathbb{R}}} \frac{1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}(f)]} \frac{\text{NJ}_{(f,e_0)}\pi_1}{\text{NJ}_{(f,e_0)}\pi_2} dV_{e_0}^{\mathbb{R}}.$$

□

Nota 2.0.15. *Nótese que si $(d) = (d_1, \dots, d_m)$ es un vector de números naturales impares entonces $\nu_{\Delta}[\mathcal{R}_{(d)}] = \nu_{\Delta}[\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})]$ (véase la proposición 1.2.2) y el lema anterior quedaría en los siguientes términos:*

$$\int_{V_{e_0}^{\mathbb{R}}} \frac{1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}(f)]} \frac{\text{NJ}_{(f,x)}\pi_1}{\text{NJ}_{(f,x)}\pi_2} dV_{e_0}^{\mathbb{R}} = \frac{\vartheta_N}{\vartheta_n}.$$

Capítulo 3

Norma media de los ceros de un sistema de ecuaciones

Algebra is the offer made by the devil to the mathematician. The devil says: «I will give you this powerful machine, it will answer any question you like. All you need to do is give me your soul: give up geometry and you will have this marvelous machine»¹.

Sir Michael Atiyah [Ati02]

En este capítulo estudiaremos el comportamiento de la norma de las soluciones de un sistema de ecuaciones polinomiales multivariadas. Es decir, en este capítulo demostraremos los Teoremas 0.0.32, 0.0.33 y 0.0.34

3.1 Notaciones preliminares

Comenzamos introduciendo algunas notaciones necesarias acerca de las cartas afines del espacio proyectivo $\mathbb{P}_n(\mathbb{R})$. Para todo número natural i tal que $0 \leq i \leq n$, definimos el conjunto abierto y denso $\mathbb{A}_i^{\mathbb{R}} \subset \mathbb{P}_n(\mathbb{R})$ como el conjunto de puntos proyectivos cuyas coordenadas homogéneas i -ésimas sean distintas de 0. Es decir,

$$\mathbb{A}_i^{\mathbb{R}} := \{x = (x_0 : \dots : x_n) \in \mathbb{P}_n(\mathbb{R}) : x_i \neq 0\}.$$

El complementario del conjunto $\mathbb{A}_i^{\mathbb{R}}$ en $\mathbb{P}_n(\mathbb{R})$ es el hiperplano $H_i = \mathbb{P}_n(\mathbb{R}) \setminus \mathbb{A}_i^{\mathbb{R}}$. Para todo número natural i tal que $0 \leq i \leq n$ tenemos las cartas afines

$$\varphi_i : \mathbb{R}^n \longrightarrow \mathbb{A}_i^{\mathbb{R}} \subset \mathbb{P}_n(\mathbb{R})$$

dadas por las siguientes igualdades: para todo $x = (x_1, \dots, x_n) \in \mathbb{R}^n$

$$\varphi_i(x) := (x_1 : \dots : x_{i-1} : 1 : x_i : \dots, x_n).$$

¹El álgebra es la oferta hecha por el diablo al matemático. El diablo dijo: «Te daré esta potente máquina que responderá cualquier cuestión. Todo lo que necesitas es darme tu alma: deja la geometría y te la daré.»

3. Norma media de los ceros de un sistema de ecuaciones

Denotamos por $e_i \in \mathbb{A}_i^{\mathbb{R}}$ al punto proyectivo dado por $e_i = \varphi_i(0) \in \mathbb{A}_i^{\mathbb{R}}$ donde $0 \in \mathbb{R}^n$ denota el origen. Nótese que para todo número natural i tal que $0 \leq i \leq n$, φ_i es una isometría en $0 \in \mathbb{R}^n$. En particular $\text{NJ}_0\varphi_i = 1$ para todo $i, 0 \leq i \leq n$.

El siguiente lema generaliza a coordenadas reales el Lema 21 de [BP07a].

Lema 3.1.1. *Usando la notación anterior. Para todo $x \in \mathbb{R}^n$ y para todo $i, 0 \leq i \leq n$ se tiene:*

$$\text{NJ}_x\varphi_i = \frac{1}{(1 + \|x\|^2)^{\frac{n+1}{2}}},$$

donde $\text{NJ}_x\varphi_i$ denota el jacobiano normal de φ_i en $x \in \mathbb{R}^n$.

Demostración. Como la prueba es idéntica para todo $i, 1 \leq i \leq n$ desarrollaremos solo la prueba para $i = 0$. Aprovechando esto, simplificaremos la notación escribiendo φ en vez de φ_1 y $e = e_1 = (1 : \dots : 0)$.

Como ya hemos observado $d_0\varphi : T_0\mathbb{R}^n \rightarrow T_e\mathbb{P}_n(\mathbb{R})$ es un isomorfismo lineal y una isometría y por lo tanto

$$\text{NJ}_0\varphi = 1.$$

Por otro lado, sea $\mathcal{O}(n+1, \mathbb{R}) \subseteq \mathcal{M}_{n+1}(\mathbb{R})$ el grupo ortogonal. Como ya hemos denotado anteriormente, tomaremos $\pi : \mathbb{R}^{n+1} \setminus \{0\} \rightarrow \mathbb{P}_n(\mathbb{R})$ como la proyección canónica. Para toda matriz ortogonal $U \in \mathcal{O}(n+1, \mathbb{R})$ abusando de la notación escribiremos U para la isometría afín $U : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^{n+1}$ dada por:

$$U(x) := Ux^t \text{ para todo } x \in \mathbb{R}^{n+1},$$

donde x^t denota el vector traspuesto de x . Denotaremos por \tilde{U} a la única isometría $\tilde{U} : \mathbb{P}_n(\mathbb{R}) \rightarrow \mathbb{P}_n(\mathbb{R})$ que hace al siguiente diagrama conmutativo

$$\begin{array}{ccc} \mathbb{R}^{n+1} \setminus \{0\} & \xrightarrow{U} & \mathbb{R}^{n+1} \setminus \{0\} \\ \pi \downarrow & & \downarrow \pi \\ \mathbb{P}_n(\mathbb{R}) & \xrightarrow{\tilde{U}} & \mathbb{P}_n(\mathbb{R}) \end{array}$$

Para todo $x \in \mathbb{R}^n$ existe una matriz ortogonal $U \in \mathcal{O}(n+1, \mathbb{R})$ tal que

$$\tilde{U}(\varphi(x)) = e = (1 : 0 : \dots : 0).$$

Nótese que es posible escoger dicha matriz U de la siguiente manera:

$$U \begin{pmatrix} 1 \\ x^t \end{pmatrix} = \begin{pmatrix} (1 + \|x\|^2)^{1/2} \\ 0 \end{pmatrix}.$$

Se tiene entonces que \tilde{U} es una isometría en cualquier punto proyectivo $x \in \mathbb{P}_n(\mathbb{R})$ y por lo tanto su jacobiano normal cumple

$$\text{NJ}_x\tilde{U} = 1, \forall x \in \mathbb{P}_n(\mathbb{R}).$$

Definimos ahora $\phi : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ como:

$$\phi := \varphi^{-1} \circ \tilde{U} \circ \varphi.$$

Nótese que $\phi(x) = 0$ y $\varphi \circ \phi = U \circ \varphi$. Usando la *regla de la cadena* se obtiene

$$\text{NJ}_0 \varphi \text{NJ}_x \phi = \text{NJ}_{\varphi(x)} \tilde{U} \text{NJ}_x \varphi.$$

Como φ es una isometría en el origen $0 \in \mathbb{R}^n$ y \tilde{U} es una isometría en cualquier punto, inmediatamente tenemos:

$$\text{NJ}_x \phi = \text{NJ}_x \varphi.$$

Es más, si $g : \mathbb{R}^n \longrightarrow \mathbb{R}$ es la aplicación

$$g(x) := \langle U_0, (1, x) \rangle, \forall x \in \mathbb{R}^n,$$

donde $\langle \cdot, \cdot \rangle$ denota el producto escalar euclídeo en \mathbb{R}^{n+1} , se tiene que

$$(g \cdot \phi)(x) = (\langle U_1, (1, x) \rangle, \dots, \langle U_n, (1, x) \rangle),$$

para todo $x \in \mathbb{R}^n$. Dado que $\phi(x) = 0$ concluimos inmediatamente que para todo $v \in T_x \mathbb{R}^n$,

$$g(x) d_x \phi(v) = (\langle U_1, (1, v) \rangle, \dots, \langle U_n, (1, v) \rangle)$$

y por lo tanto

$$d_x \phi(v) = \frac{1}{(1 + \|x\|^2)^{1/2}} (\langle U_1, (1, v) \rangle, \dots, \langle U_n, (1, v) \rangle),$$

para todo $v \in T_x \mathbb{R}^n$.

Introducimos ahora un lema técnico que usaremos a continuación.

Lema 3.1.2. *Para todo vector tangente $v \in T_x \mathbb{R}^n$ tal que $\langle x, v \rangle_{\mathbb{R}} = 0$ y para cualquier otro vector tangente $w \in T_x \mathbb{R}^n$ se cumple:*

$$\langle d_x \phi(v), d_x \phi(w) \rangle_{T_0 \mathbb{R}^n} = \frac{1}{1 + \|x\|_2^2} \langle v, w \rangle_{\mathbb{R}}.$$

Demostración. Nótese que para todo $v, w \in T_x \mathbb{R}^n$ se tiene:

$$\langle d_x \phi(v), d_x \phi(w) \rangle_{T_0 \mathbb{R}^n} = \frac{1}{1 + \|x\|_2^2} \sum_{i=1}^n \langle U_i, (0, v) \rangle_{\mathbb{R}} \langle U_i, (0, w) \rangle_{\mathbb{R}}.$$

Ahora bien, como

$$\left\langle U \begin{pmatrix} 1 \\ v^t \end{pmatrix}, U \begin{pmatrix} 1 \\ w^t \end{pmatrix} \right\rangle_{\mathbb{R}} = \langle v, w \rangle_{\mathbb{R}},$$

concluimos inmediatamente:

$$\langle d_x \phi(v), d_x \phi(w) \rangle_{T_0 \mathbb{R}^n} = \frac{1}{1 + \|x\|_2^2} (\langle v, w \rangle_{\mathbb{R}} - \langle U_0, (0, v) \rangle \langle U_0, (0, w) \rangle).$$

Como $\langle x, v \rangle_{\mathbb{R}} = 0$ se tiene que $\langle U_0, (0, v) \rangle = 0$ y se sigue el lema. \square

3. Norma media de los ceros de un sistema de ecuaciones

Consideramos ahora una base ortonormal $\{b_1, \dots, b_n\}$ de $T_x\mathbb{R}^n$ tal que $b_n = \frac{x}{\|x\|}$. Tenemos entonces que, para todo $i, 1 \leq i \leq n-1$ y para todo $j, 1 \leq j \leq n$ aplicando el Lema 3.1.2 se cumple:

$$\langle d_x\phi(v), d_x\phi(w) \rangle_{T_0\mathbb{R}^n} = \frac{1}{1 + \|x\|_2^2} \langle b_i, b_j \rangle_{T_x\mathbb{R}^n} = \begin{cases} \frac{1}{1 + \|x\|_2^2} & \text{si } i = j \\ 0 & \text{en otro caso.} \end{cases}$$

Y para $i = n$ se tiene que:

$$\langle d_x\phi(v), d_x\phi(w) \rangle_{T_0\mathbb{R}^n} = \frac{1}{1 + \|x\|_2^2} \left[1 - \frac{1}{\|x\|_2^2} \langle U_0, (0, x) \rangle^2 \right].$$

Obsérvese que

$$\langle U_0, (0, x) \rangle^2 = \frac{\|x\|_2^4}{1 + \|x\|_2^2},$$

por lo tanto

$$\langle d_x\phi(v), d_x\phi(w) \rangle_{T_0\mathbb{R}^n} = \frac{1}{1 + \|x\|_2^2} \left[1 - \frac{\|x\|_2^2}{1 + \|x\|_2^2} \right] = \frac{1}{(1 + \|x\|_2^2)^2}.$$

Finalmente

$$\text{NJ}_x\varphi \text{NJ}_x\phi = |\det W(\phi, x)|^{1/2} = \left(\frac{1}{1 + \|x\|_2^2} \right)^{\frac{n+1}{2}},$$

donde $W(\phi, x)$ es la matriz simétrica dada por

$$W(\phi, x) = (\langle d_x\phi(b_i), d_x\phi(b_j) \rangle)_{1 \leq i, j \leq n}.$$

Y el enunciado se sigue. □

Proposición 3.1.3. *Para todo número real r tal que $0 < r < 1$ y para todo número natural i tal que $0 \leq i \leq n$ se cumple:*

$$\frac{1}{\vartheta_n} \int_{\mathbb{P}_n(\mathbb{R})} \|\varphi_i^{-1}(x)\|^r d\nu_{\mathbb{P}} = \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right).$$

donde $\beta(x, y)$ representa la función especial beta (véase Anexo A).

Demostración. Como en el caso anterior sólo desarrollaremos el caso $i = 0$. También simplificaremos la notación de la misma manera, es decir denotaremos $\varphi := \varphi_0$.

Como $\mathbb{P}_n(\mathbb{R}) \setminus \mathbb{A}_0^{\mathbb{R}}$ es un hiperplano y por lo tanto tiene medida nula, podemos usar la fórmula de la coárea (véase Teorema 1.3.3) para concluir:

$$I(r) := \int_{z \in \mathbb{P}_n(\mathbb{R})} \|\varphi^{-1}(z)\|^r d\nu_{\mathbb{P}} = \int_{x \in \mathbb{R}^n} \int_{z \in \varphi(x)} \frac{\|x\|^r}{\text{NJ}_z\varphi^{-1}} d\varphi(x) d\mathbb{R}^n.$$

Aplicando los Lemas 3.1.1 y 1.3.2 se sigue que

$$I(r) = \int_{x \in \mathbb{R}^n} \frac{\|x\|^r}{(1 + \|x\|^2)^{\frac{n+1}{2}}} d\mathbb{R}^n.$$

Transformamos ahora la integral a coordenadas esféricas. Para ello definimos la siguiente aplicación diferencial:

$$\rho : \begin{array}{ccc} (0, +\infty) \times \mathbb{S}_{n-1}(\mathbb{R}) & \longrightarrow & \mathbb{R}^n \setminus \{0\} \\ (t, v) & \longmapsto & tv \end{array},$$

que tiene por inversa:

$$\rho^{-1} : \begin{array}{ccc} \mathbb{R}^n \setminus \{0\} & \longrightarrow & (0, +\infty) \times \mathbb{S}_{n-1}(\mathbb{R}) \\ x & \longmapsto & (\|x\|, \frac{x}{\|x\|}) \end{array}.$$

Los jacobianos normales de ρ y ρ^{-1} cumplen:

$$\begin{aligned} \text{NJ}_x \rho^{-1} &= \frac{1}{\|x\|^{n-1}} \quad \forall x \in \mathbb{R}^n \setminus \{0\} \\ \text{NJ}_x \rho &= t^{n-1} \quad \forall (t, v) \in (0, +\infty) \times \mathbb{S}_{n-1}(\mathbb{R}). \end{aligned}$$

Aplicando, de nuevo, la fórmula de la coárea obtenemos:

$$I(r) = \int_{(t,v) \in (0,+\infty) \times \mathbb{S}_{n-1}(\mathbb{R})} \int_{x \in \rho(t,v)} \frac{\|x\|^r}{(1 + \|x\|^2)^{\frac{n+1}{2}}} \frac{1}{\text{NJ}_x \rho^{-1}} d\rho(t, v) d(\mathbb{R} \times \mathbb{S}_{n-1}(\mathbb{R})).$$

Lo cual da a lugar:

$$I(r) = \nu[\mathbb{S}_{n-1}(\mathbb{R})] \int_0^{+\infty} \frac{t^{n+r-1}}{(1+t^2)^{\frac{n+1}{2}}} dt.$$

Usando la siguiente igualdad (véase Anexo A)

$$\beta(x, y) = 2 \int_0^{\infty} \frac{t^{2x-1}}{(1+t^2)^{x+y}} dt,$$

se obtiene

$$I(r) = \frac{\nu[\mathbb{S}_{n-1}(\mathbb{R})]}{2} \beta\left(\frac{n+r}{2}, \frac{1-r}{2}\right).$$

El resultado se sigue teniendo en cuenta que

$$\vartheta_n = \frac{\nu[\mathbb{S}_{n-1}(\mathbb{R})]}{2}.$$

□

Estamos en condiciones de demostrar los teoremas principales de este capítulo.

3.2 Demostración del Teorema 0.0.32

Por definición tenemos que:

$$\vartheta_N E_\Delta [N_{av}^r(f)] = \int_{\mathcal{R}(d)} \frac{1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]} \left(\int_{x \in \mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)} \|\varphi_0^{-1}(x)\|^r dV_{\mathbb{P}}^{\mathbb{R}}(f) \right) d\nu_\Delta.$$

Como $\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f) = \pi_1^{-1}(f)$, usando la fórmula de la coárea obtenemos

$$\begin{aligned} \vartheta_N E_\Delta [N_{av}^r(f)] &= \int_{(f,x) \in V^{\mathbb{R}}} \frac{1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]} \|\varphi_0^{-1}(x)\|^r \text{NJ}_{(f,x)} \pi_1 dV = \\ &= \int_{x \in \mathbb{P}_n(\mathbb{R})} \int_{f \in (\mathcal{R}(d))_x} \frac{1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]} \|\varphi_0^{-1}(x)\|^r \frac{\text{NJ}_{(f,x)} \pi_1}{\text{NJ}_{(f,x)} \pi_2} dV_x^{\mathbb{R}} d\nu_n, \end{aligned}$$

donde $(\mathcal{R}(d))_x = \mathcal{R}(d) \cap \pi_2^{-1}(x) = V_x$. Obsérvese que la siguiente integral no depende de x

$$\int_{(f,x) \in V_x} \frac{1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]} \frac{\text{NJ}_{(f,x)} \pi_1}{\text{NJ}_{(f,x)} \pi_2} dV_x^{\mathbb{R}}.$$

Luego, obtenemos la siguiente igualdad

$$\vartheta_N E_\Delta [N_{av}^r(f)] = \left(\int_{x \in \mathbb{P}_n(\mathbb{R})} \|\varphi_0^{-1}(x)\|^r d\nu_n \right) \left(\int_{(f,e_0) \in V_{e_0}} \frac{1}{\nu_{n-m}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]} \frac{\text{NJ}_{(f,e_0)} \pi_1}{\text{NJ}_{(f,e_0)} \pi_2} dV_{e_0}^{\mathbb{R}} \right).$$

Ahora, aplicando el Lema 2.0.14 y el Lema 3.1.3, conseguimos

$$\vartheta_N E_\Delta [N_{av}^r(f)] = \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right) \nu_\Delta[\mathcal{R}(d)],$$

tal como queríamos. □

A partir de este teorema se pueden obtener los siguientes corolarios:

Corolario 3.2.1. *Usando la notación anterior*

$$\text{Prob}_\Delta [N_{av}^r(f) \geq \varepsilon^{-1}] \leq \frac{\nu_\Delta[\mathcal{R}(d)]}{\vartheta_N} \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right) \varepsilon.$$

En particular, si todos los elementos de la lista de grados $(d) = (d_1, \dots, d_m)$ son números impares se tiene

$$\text{Prob}_\Delta [N_{av}^r(f) \geq \varepsilon^{-1}] \leq \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right) \varepsilon.$$

Demostración. Simplemente aplicar la *Desigualdad de Markov* (véase 1.3.6) al teorema anterior. □

Corolario 3.2.2. Usando la notación del Teorema 0.0.32 anterior. Para todo ε número real positivo tal que $\varepsilon < 1$ la siguiente desigualdad se cumple

$$\text{Prob}_\Delta [\dim(\mathbb{V}_\mathbb{A}^\mathbb{R}(f) \cap \overline{B}(0, \varepsilon^{-2})) < n - m] \leq \varepsilon \beta \left(\frac{n}{2} + \frac{1}{4}, \frac{1}{4} \right) \frac{\nu_\Delta[\mathcal{R}]}{\vartheta_N},$$

donde $\dim(\cdot)$ denota la dimensión como conjunto semialgebraico y $\overline{B}(0, \varepsilon^{-2})$ denota la bola cerrada de centro 0 y radio ε^{-2} en \mathbb{R}^n . En particular, si todos los elementos de la lista de grados $(d) = (d_1, \dots, d_m)$ son números impares, $n \geq 3$ y ε es un número real tal que $\varepsilon < 1$, entonces

$$\text{Prob}_\Delta [\dim(\mathbb{V}_\mathbb{A}^\mathbb{R}(f) \cap \overline{B}(0, \varepsilon^{-2})) < n - m] \leq \varepsilon \Gamma \left(\frac{1}{4} \right).$$

Demostración. Para todo número real ε tal que $\varepsilon > 0$, para casi todo f en $\mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R})$, se tiene que la dimensión del conjunto

$$\mathbb{V}_\mathbb{A}^\mathbb{R}(f) \cap B(0, \varepsilon^{-2}),$$

es menor que $n - m$ si y solo si dicho conjunto tiene medida nula como subconjunto de $\mathbb{V}_\mathbb{A}^\mathbb{R}(f)$.

En el caso que $\mathbb{V}_\mathbb{A}^\mathbb{R}(f) \cap B(0, \varepsilon^{-2})$ tenga medida nula en $\mathbb{V}_\mathbb{A}^\mathbb{R}(f)$, se tiene para casi todo x en $\mathbb{V}_\mathbb{A}^\mathbb{R}(f)$ que:

$$\|x\| > \varepsilon^{-2}.$$

De lo que se deduce que para casi todo x en $\mathbb{V}_\mathbb{A}^\mathbb{R}(f)$:

$$\|x\|^{1/2} > \varepsilon^{-1},$$

y por lo tanto $N_{av}^{1/2}(f) \geq \varepsilon^{-1}$. Finalmente

$$\begin{aligned} \text{Prob}_\Delta [f \in \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R}) : \dim(\mathbb{V}_\mathbb{A}^\mathbb{R}(f) \cap B(0, \varepsilon^{-1})) < n - m] &\leq \\ &\leq \text{Prob}_\Delta [f \in \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R}) : N_{av}^{1/2} \geq \varepsilon^{-1}]. \end{aligned}$$

Aplicando la *Desigualdad de Markov* (véase 1.3.6) concluimos

$$\text{Prob}_\Delta [f \in \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R}) : \dim(\mathbb{V}_\mathbb{A}^\mathbb{R}(f) \cap B(0, \varepsilon^{-1})) < n - m] \leq \varepsilon E_\Delta [N_{av}^{1/2}(f)],$$

y el resultado se sigue aplicando el enunciado del Teorema 0.0.32. □

3.3 Demostración del Teorema 0.0.33

Para todo $f \in \mathbb{P}(\mathcal{H}_{(d)}^\mathbb{R})$ definimos $\mathbb{V}_\mathbb{A}^\mathbb{C}(f) \subseteq \mathbb{C}^n$ el conjunto algebraico de todas las soluciones afines sobre los números complejos del sistema de ecuaciones definido por f . Es decir

$$\mathbb{V}_\mathbb{A}^\mathbb{C}(f) := \{x \in \mathbb{C}^n : f_i(x) = 0, 1 \leq i \leq m\},$$

3. Norma media de los ceros de un sistema de ecuaciones

donde $f = (f_1 \dots, f_m)$. Sea $U_\infty \subseteq \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ el conjunto de los sistemas $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ tales que $V_{\mathbb{A}}^{\mathbb{C}}(f)$ es una variedad algebraica compleja de dimensión menor que $n - m$. A partir de resultados clásicos de *Teoría de la Eliminación* se tiene que U_∞ es un subconjunto propio de una subvariedad algebraica de $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$. En particular, la siguiente igualdad se cumple en $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$:

$$\nu_N^{\mathbb{R}}[U_\infty] = 0.$$

Ahora, para todo $\alpha \in \{0, 1\}$ consideramos la siguiente integral

$$I^\alpha(r) = \int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \int_{x \in V_{\mathbb{P}}^{\mathbb{R}}(f)} \|\varphi_0^{-1}(x)\|^{r\alpha} dV_{\mathbb{P}}^{\mathbb{R}}(f) d\nu_\Delta.$$

Es más, para todo $f \in \mathcal{H}_{(d)}^{\mathbb{R}} \setminus U_\infty$ se tiene que

$$\mathbb{E}_{\mathcal{H}_{(d)}^{\mathbb{R}}} \left[\int_{x \in V_{\mathbb{A}}^{\mathbb{R}}(f)} \|x\|^r dV_{\mathbb{A}}^{\mathbb{R}}(f) \right] = \frac{I^1(r)}{\vartheta_N}. \quad (3.1)$$

Nuestro objetivo es calcular $I^\alpha(r)$ pues el resultado se sigue de forma inmediata. Usando la fórmula de la coárea (véase 1.3.3) obtenemos:

$$I^\alpha(r) = \int_{x \in \mathbb{P}_n(\mathbb{R})} \int_{(f,x) \in V_x^{\mathbb{R}}} \|\varphi_0^{-1}(x)\|^{r\alpha} \frac{\text{NJ}_{(f,x)}\pi_1}{\text{NJ}_{(f,x)}\pi_2} dV_x^{\mathbb{R}} d\nu_{\mathbb{P}}.$$

Claramente se tiene que

$$\frac{\text{NJ}_{(f,x)}\pi_1}{\text{NJ}_{(f,x)}\pi_2} dV_x^{\mathbb{R}},$$

es independiente de x , luego

$$I^\alpha(r) = \int_{x \in \mathbb{P}_n(\mathbb{R})} \|\varphi_0^{-1}(x)\|^{r\alpha} d\nu_{\mathbb{P}} \int_{f \in V_{e_0}^{\mathbb{R}}} \frac{\text{NJ}_{(f,e_0)}\pi_1}{\text{NJ}_{(f,e_0)}\pi_2} dV_{e_0}^{\mathbb{R}}.$$

Consideramos ahora el caso $\alpha = 0$. Aplicando el Teorema 0.0.35 obtenemos:

$$I^0(r) = \int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \int_{x \in V_{\mathbb{P}}^{\mathbb{R}}(f)} dV_{\mathbb{P}}^{\mathbb{R}}(f) d\nu_\Delta = \mathcal{D}^{1/2} \vartheta_{n-m} \vartheta_N.$$

Por otro lado,

$$I^0(r) = \vartheta_n \int_{f \in V_{e_0}^{\mathbb{R}}} \frac{\text{NJ}_{(f,e_0)}\pi_1}{\text{NJ}_{(f,e_0)}\pi_2} dV_{e_0}^{\mathbb{R}}.$$

Juntando ambas igualdades tenemos

$$\int_{f \in V_{e_0}^{\mathbb{R}}} \frac{\text{NJ}_{(f,e_0)}\pi_1}{\text{NJ}_{(f,e_0)}\pi_2} dV_{e_0}^{\mathbb{R}} = \mathcal{D}^{1/2} \frac{\vartheta_{n-m} \vartheta_N}{\vartheta_n}. \quad (3.2)$$

Consideramos ahora el caso $\alpha = 1$. Usando la Ecuación (3.2) y el Teorema 3.1.3 tenemos

$$I^1(r) = \mathcal{D}^{1/2} \vartheta_{n-m} \vartheta_N \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right).$$

Finalmente, usando la Igualdad (3.1) concluimos

$$E_{\Delta} \left[\int_{x \in V_{\mathbb{A}}^{\mathbb{R}}(f)} \|x\|^r dV_{\mathbb{A}}^{\mathbb{R}}(f) \right] = \mathcal{D}^{1/2} \vartheta_{n-m} \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right),$$

□

Demostrar un corolario necesario para demostrar la demostración del Corolario 0.0.34.

Corolario 3.3.1. *Usando las notaciones del Teorema 0.0.32. Si $n = m$, $n \geq 3$ y para todo número real r tal que $0 < r < 1$ se tiene:*

$$\text{Prob}_{\Delta} \left[\max_{x \in V_{\mathbb{A}}^{\mathbb{R}}(f)} \|x\| \geq \varepsilon^{-1} \right] \leq \mathcal{D}^{1/2} \varepsilon^r \Gamma \left(\frac{1-r}{2} \right),$$

donde Γ es la función Γ definida por Euler y \mathcal{D} denota el número de Bézout, es decir $\mathcal{D} = \prod_{i=1}^m d_i$.

Demostración. De forma trivial tenemos que para todo número real r tal que $0 < r < 1$:

$$\text{Prob}_{\Delta} \left[\max_{x \in V_{\mathbb{A}}^{\mathbb{R}}(f)} \|x\| \geq \varepsilon^{-1} \right] = \text{Prob}_{\Delta} \left[\max_{x \in V_{\mathbb{A}}^{\mathbb{R}}(f)} \|x\|^r \geq \varepsilon^{-r} \right].$$

La siguiente afirmación también es trivial:

$$\text{Prob}_{\Delta} \left[\max_{x \in V_{\mathbb{A}}^{\mathbb{R}}(f)} \|x\|^r \geq \varepsilon^{-r} \right] \leq \text{Prob}_{\Delta} \left[\sum_{x \in V_{\mathbb{A}}^{\mathbb{R}}(f)} \|x\|^r \geq \varepsilon^{-r} \right].$$

Finalmente, usando la *Desigualdad de Markov* (1.3.6) y el Lema 0.0.33 (en el caso $n = m$) obteniendo:

$$\text{Prob}_{\Delta} \left[\sum_{x \in V_{\mathbb{A}}^{\mathbb{R}}(f)} \|x\|^r \geq \varepsilon^{-r} \right] \leq \mathcal{D}^{1/2} \varepsilon^r \beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right).$$

Por otro lado, es fácil probar que si $n \geq 3$ y para todo número real r tal que $0 < r < 1$ se cumple:

$$\beta \left(\frac{n+r}{2}, \frac{1-r}{2} \right) \leq \Gamma \left(\frac{1-r}{2} \right).$$

Y sustituyendo se sigue el resultado. □

El demostración del Corolario 0.0.34 es inmediata a partir del resultado anterior.

Nota 3.3.2. *Es importante remarcar el significado del Corolario 0.0.34 La bola cerrada $\overline{B}(0, \varepsilon^{-2})$ contiene un conjunto denso en la topología Zariski de alguna de las componente irreducibles de $V_{\mathbb{A}}^{\mathbb{R}}(f)$ con alta probabilidad.*

3.4 Número de soluciones reales de un sistema de ecuaciones

El Teorema 0.0.35 extiende al caso subdeterminado el resultado acerca del número de soluciones reales que presenta un sistema de ecuaciones polinomiales multivariadas en el caso cero dimensional debido a M. Shub y S. Smale (véase [BCSS98]). Kostlan demostró en [Kos93] (véase también [Kos02]) el caso en el que todos los grados de los polinomios sean iguales. La formulación siguiente es debida a Prior y Bürguisser (véase [Pri06]). Se presenta aquí debido a que es un mero corolario de nuestras técnicas.

Demostración del Teorema 0.0.35. Sea $\rho : B^1(L_{e_0}) \longrightarrow B^1(\mathcal{M}_{m \times n}(\mathbb{R}))$ la isometría:

$$\rho(f) := \Delta((d)^{1/2})T_{e_0}f,$$

donde $B^1(\mathcal{M}_{m \times n}(\mathbb{R}))$ es la bola cerrada en $\mathcal{M}_{m \times n}(\mathbb{R})$ respecto de la norma de Frobenius. Tenemos entonces:

$$E_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} [\nu_{\Delta}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]] = 2 \frac{\nu_n \nu_{N-(n+1)m}}{\nu_N} I(n, m, (d)),$$

donde

$$\begin{aligned} I(n, m, (d)) &= \int_{f \in B^1(L_{e_0})} |\det((T_{e_0}f)(T_{e_0}f)^t)|^{1/2} (1 - \|f\|_{\Delta}^2)^{\frac{N-(n+1)m-1}{2}} dB^1 = \\ &= \mathcal{D}^{1/2} \int_{M \in B^1(\mathcal{M}_{m \times n}(\mathbb{R}))} |\det(MM^t)|^{1/2} (1 - \|M\|_F^2)^{\frac{N-(n+1)m-1}{2}} dB^1 \end{aligned}$$

Transformamos ahora la integral a coordenadas esféricas tal como hicimos en la demostración del Lema 3.1.3 obteniendo:

$$I(n, m, (d)) = \mathcal{D}^{1/2} \int_{\mathbb{S}(\mathcal{M}_{m \times n}(\mathbb{R}))} \int_0^1 |\det((rM)(rM)^t)|^{1/2} r^{mn-1} (1 - \|rM\|_F^2)^{\frac{N-(n+1)m-1}{2}} d\mathbb{S}.$$

Tenemos entonces que

$$I(n, m, (d)) = \mathcal{D}^{1/2} J(n, m) \int_0^1 r^{(n+1)m-1} (1 - r^2)^{\frac{N-(n+1)m-1}{2}} dr,$$

donde

$$J(n, m) = \int_{\mathbb{S}(\mathcal{M}_{m \times n}(\mathbb{R}))} \det(MM^t) d\mathbb{S}.$$

Por otro lado, tal como también hemos visto en la demostración del Lema 3.1.3,

$$\beta(x, y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} = 2 \int_0^1 t^{2x-1} (1-t^2)^{y-1} dt.$$

Por lo tanto

$$\mathbb{E}_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} [\nu_{\Delta}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]] = \mathcal{D}^{1/2} \frac{\nu_n \nu_{N-(n+1)m}}{\nu_N} \beta \left(\frac{(n+1)m}{2}, \frac{N-(n+1)m+1}{2} \right) J(m, n).$$

Como $\nu_k = \frac{\pi^{\frac{k+1}{2}}}{\Gamma(\frac{k+1}{2})}$ concluimos

$$\mathbb{E}_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} [\nu_{\Delta}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]] = \mathcal{D}^{1/2} \pi^{\frac{(n+1)(1-m)}{2}} \frac{\Gamma\left(\frac{(n+1)m}{2}\right)}{\Gamma\left(\frac{n+1}{2}\right)} J(n, m).$$

Además, si denotamos por

$$R(n, m) = \pi^{\frac{(n+1)(1-m)}{2}} \frac{\Gamma\left(\frac{(n+1)m}{2}\right)}{\Gamma\left(\frac{n+1}{2}\right)} J(n, m),$$

para toda lista de grados $(d) = (d_1, \dots, d_m)$ se cumple:

$$\mathbb{E}_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} [\nu_{\Delta}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]] = \mathcal{D}^{1/2} R(n, m).$$

En particular, si $(d) = (1, \dots, 1)$, $\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)$ es un espacio proyectivo de codimensión m excepto en un conjunto de medida nula en $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ por lo que:

$$\mathbb{E}_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} [\nu_{\Delta}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]] = \nu_{n-m}.$$

Luego se sigue que

$$R(n, m) = \nu_{n-m},$$

y concluimos

$$\mathbb{E} [\nu_{\Delta}[\mathbb{V}_{\mathbb{P}}^{\mathbb{R}}(f)]] = \mathcal{D}^{1/2} \nu_{n-m}.$$

□

A partir de la demostración anterior se sigue un corolario que no tiene relación directa con nuestro discurso pero que tiene interés por si mismo.

Corolario 3.4.1. *Usando la notación anterior:*

$$\int_{\mathbb{S}(\mathcal{M}_{m \times n}(\mathbb{R}))} \det(MM^t) d\mathbb{S} = \nu_{n-m} \pi^{\frac{(n+1)(m-1)}{2}} \frac{\Gamma\left(\frac{n+1}{2}\right)}{\Gamma\left(\frac{(n+1)m}{2}\right)}$$

Este corolario nos da el valor esperado para el determinante de matrices simétricas cuyas entradas son todas ellas variables aleatorias normales de media 0 y varianza 1.

Capítulo 4

Estimación de la distribución de probabilidad del número de condicionamiento

...je n'aurai rien à y faire que de vous indiquer le moyen de répondre à votre question, sans vouloir charger ni moi ni personne de le faire. En un mot, les calculs sont impraticables¹.

Évariste Galois [GL46]

Como hemos visto en la introducción es sumamente importante conocer la distribución de probabilidad del condicionamiento μ_{norm} para conocer la complejidad de un algoritmo de deformación homotópica basado en el esquema de Shub y Smale. A este respecto dedicaremos este capítulo a demostrar los Teoremas 0.0.16 y 0.0.17. Pasamos ahora a demostrar dichos resultados.

4.1 Demostración del Teorema 0.0.16

Definimos $\chi_\varepsilon : V^{\mathbb{R}} \rightarrow \{0, 1\}$ como la función característica del siguiente conjunto

$$A_\varepsilon = \{(f, \zeta) \in V^{\mathbb{R}} : \mu_{norm}(f, \zeta) \geq \varepsilon^{-1}\}.$$

Calcularemos cotas inferiores y superiores de la siguiente cantidad:

$$I = \int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} Vol_\varepsilon^m(f) \vartheta_\Delta = \int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \left(\int_{x \in V_{\mathbb{P}}^{\mathbb{R}}(f)} \chi_\varepsilon(f, x) dV_{\mathbb{P}}^{\mathbb{R}}(f) \right) d\nu_\Delta.$$

Como el μ_{norm}^m es invariante respecto a la acción del grupo $\mathcal{O}(n+1, \mathbb{R})$ sobre $V^{\mathbb{R}}$ (véase la Proposición 1.1.1), aplicando la Proposición 2.0.13 concluimos:

$$I = 2\vartheta_n \vartheta_{N-(n+1)m-1} J(\varepsilon, n, m, (d)),$$

¹... sólo necesito indicarte el método para responder a tus preguntas, sin esperar que yo u otra persona pueda llevarlo a cabo. En resumen, los cálculos son imposibles.

4. Distribución de probabilidad del número de condicionamiento

donde

$$J(\varepsilon, n, m, (d)) = \int_{f \in B^1(L_{e_0})} \frac{\chi_\varepsilon(f, e_0)}{\|f\|_\Delta} |\det((T_{e_0}f)(T_{e_0}f)^t)|^{1/2} (1 - \|f\|_\Delta^2)^{\frac{N-(n+1)m-2}{2}} dB^1$$

y $B^1(L_{e_0})$ es la bola unitaria en L_{e_0} respecto de la norma $\|\cdot\|_\Delta$.

Definimos $\rho : B^1(L_{e_0}) \longrightarrow B^1(\mathcal{M}_{n \times m}(\mathbb{R}))$ a la isometría dada por la siguiente igualdad:

$$\rho(f) := \Delta((d)^{1/2})T_{e_0}f,$$

Tenemos entonces

$$J(\varepsilon, n, m, (d)) = \mathcal{D}^{1/2} \int_{f \in B^1(\mathcal{M}_{n \times m}(\mathbb{R}))} \frac{\chi'_\varepsilon(M)}{\|M\|_F} |\det(MM^t)|^{1/2} (1 - \|M\|_F^2)^{\frac{N-(n+1)m-2}{2}} dB^1,$$

donde $\chi'_\varepsilon : \mathcal{M}_{n \times m}(\mathbb{R}) \longrightarrow \{0, 1\}$ es la función característica del conjunto:

$$B_\varepsilon = \{M \in \mathcal{M}_{n \times m}(\mathbb{R}) : \kappa_D(M) \geq \varepsilon^{-1}\}.$$

Nótese que la función χ'_ε es homogénea de grado 0 y por lo tanto una función proyectiva. Transformamos ahora la integral a coordenadas esféricas tal como hicimos en la demostración del Lema 3.1.3 obteniendo:

$$\begin{aligned} J(\varepsilon, n, m, (d)) &= \\ &= \mathcal{D}^{1/2} \int_{M \in \mathbb{S}(\mathcal{M}_n(\mathbb{R}))} \int_0^1 \chi'_\varepsilon(rM) |\det((rM)(rM)^t)|^{1/2} r^{nm-2} (1 - \|rM\|_F^2)^{\frac{N-(n+1)m-2}{2}} dr d\mathbb{S} \end{aligned}$$

O equivalentemente,

$$J(\varepsilon, n, m, (d)) = H(\varepsilon, n, m) \mathcal{D}^{1/2} \int_0^1 r^{(n+1)m-2} (1 - r^2)^{\frac{N-(n+1)m-2}{2}} dr,$$

donde:

$$H(\varepsilon, n, m) = \int_{M \in \mathbb{S}(\mathcal{M}_{n \times m}(\mathbb{R}))} \chi'_\varepsilon(M) |\det(MM^t)|^{1/2} d\mathbb{S} = 2 \int_{M \in \mathbb{P}(\mathcal{M}_{n \times m}(\mathbb{R}))} \chi'_\varepsilon(M) |\det(MM^t)|^{1/2} d\nu_{\mathbb{P}}.$$

Consideramos ahora el espacio proyectivo $\mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R}))$ y la variedad de incidencia

$$W = \{(M, x) \in \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R})) \times \mathbb{P}_n(\mathbb{R}) : Mx = 0\}.$$

Nótese que

$$\mathcal{M}_{m \times (n+1)}(\mathbb{R}) = \mathcal{H}_{(1)}^{\mathbb{R}},$$

donde $(1) = (1, \dots, 1) \in \mathbb{N}^m$. Tenemos entonces que W es una variedad Riemannian y junto a las proyecciones presentadas en la introducción obtenemos el siguiente triángulo de proyecciones

$$\begin{array}{ccc} & W & \\ \pi_1^{(1)} \swarrow & & \searrow \pi_2^{(1)} \\ \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R})) & & \mathbb{P}_n(\mathbb{R}). \end{array}$$

Téngase en cuenta que $\pi_2^{(1)-1}(e_0)$ puede ser identificado con el espacio proyectivo $\mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R}))$. Es decir, $\pi_2^{(1)-1}(e_0) = W_{e_0} = \mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R}))$. Es más, para toda matriz $M \in \pi_2^{(1)-1}(e_0)$, se tiene que $T_{e_0}M = M$. Luego, usando el Lema 2.0.10 obtenemos la siguiente igualdad

$$H(\varepsilon, n, m) = \frac{2}{\vartheta_n} \int_{x \in \mathbb{P}_n(\mathbb{R})} \left(\int_{M \in \pi_2^{(1)-1}(x)} \chi_\varepsilon''(M, x) \frac{\text{NJ}_{(M,x)}\pi_1^{(1)}}{\text{NJ}_{(M,x)}\pi_2^{(1)}} d\pi_2^{(1)-1}(x) \right) d\nu_{\mathbb{P}},$$

donde $\chi_\varepsilon'' : W \rightarrow \{0, 1\}$ es la función característica del conjunto

$$D_\varepsilon = \{(M, x) \in W : \mu_{norm}(M, x) > \varepsilon^{-1}\}.$$

Seguidamente obtenemos:

$$H(\varepsilon, n, m) = \frac{2}{\vartheta_n} \int_W \chi_\varepsilon''(M, x) \text{NJ}_{(M,x)}\pi_1^{(1)} dW,$$

y finalmente:

$$H(\varepsilon, n, m) = \frac{2}{\vartheta_n} \int_{M \in \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R}))} \chi_\varepsilon'''(M) d\nu_{\mathbb{P}},$$

donde $\chi_\varepsilon''' : \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R})) \rightarrow \{0, 1\}$ es la función característica del conjunto $\pi_1^{(1)}(D_\varepsilon)$.

Ahora observamos que existe un subconjunto propio de $S \subseteq \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R}))$ cerrado en la *topología Zariski* tal que, para todo $M \in \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R})) \setminus S$ la fibra $\pi_1^{(1)-1}(M)$ es un punto proyectivo. Es decir, para todo $M \in \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R})) \setminus S$, el núcleo de M es un subespacio vectorial de \mathbb{R}^{n+1} de dimensión 1 y además M define una aplicación lineal. En particular, para todo $M \in \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R})) \setminus S$, se tiene que $\mu_{norm}(M, x) = \kappa_\Delta(M)$ y podemos concluir

$$H(\varepsilon, n, m) = \frac{2}{\vartheta_n} \int_{M \in \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R}))} \chi_\varepsilon^{iv}(M) d\nu_{\mathbb{P}},$$

donde $\chi_\varepsilon^{iv} : \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R})) \rightarrow \{0, 1\}$ es la función característica del conjunto

$$F_\varepsilon = \{M \in \mathbb{P}(\mathcal{M}_{m \times (n+1)}(\mathbb{R})) : \kappa_\Delta(M) \geq \varepsilon^{-1}\}.$$

Ahora bien, usando los resultados debidos a Chen-Dongarra (véase también el corolario 1.4.3) concluimos:

$$\frac{2}{\vartheta_n} \frac{\vartheta_{m(n+1)-1}}{\sqrt{2\pi}} \left(\frac{(n+1)c\varepsilon}{n-m+2} \right)^{n-m+2} \leq H(\varepsilon, n, m) \leq \frac{2}{\vartheta_n} \frac{\vartheta_{m(n+1)-1}}{\sqrt{2\pi}} \left(\frac{(n+1)^{3/2}C\varepsilon}{n-m+2} \right)^{n-m+2}$$

donde $0.245 \leq c \leq 2$ y $5.013 \leq C \leq 6.414$.

Por otro lado, tal como hemos visto en la demostración del Lema 3.1.3 (véase Anexo A para más detalles),

$$\beta(x, y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} = 2 \int_0^1 t^{2x-1}(1-t^2)^{y-1} dt,$$

4. Distribución de probabilidad del número de condicionamiento

se tiene que:

$$\int_0^1 r^{(n+1)m-2}(1-r^2)^{\frac{N-(n+1)m-2}{2}} dr = \beta \left(\frac{(n+1)m-1}{2}, \frac{N-(n+1)m+1}{2} \right),$$

y usando la *desigualdad de Gautchi* (véase [EGP00] por ejemplo) se obtiene fácilmente:

$$\begin{aligned} \left(\frac{2}{(n+1)m-1} \right)^{1/2} \beta \left(\frac{(n+1)m}{2}, \frac{N-(n+1)m+1}{2} \right) &\leq \\ &\leq \beta \left(\frac{(n+1)m-1}{2}, \frac{N-(n+1)m+1}{2} \right) \leq \\ &\leq \left(\frac{N}{2} \right)^{1/2} \beta \left(\frac{(n+1)m}{2}, \frac{N-(n+1)m+1}{2} \right), \end{aligned}$$

y por lo tanto:

$$\begin{aligned} \left(\frac{(n+1)m-1}{2} \right)^{-1/2} \vartheta_N &\leq \\ &\leq \vartheta_{N-(n+1)m} \vartheta_{m(n+1)-1} \beta \left(\frac{(n+1)m-1}{2}, \frac{N-(n+1)m+1}{2} \right) \leq \\ &\leq \left(\frac{N}{2} \right)^{1/2} \vartheta_N. \quad (4.1) \end{aligned}$$

Juntando todo lo anterior obtenemos:

$$I = 2\vartheta_n \vartheta_{N-(n+1)m} J(\varepsilon, n, m, (d)) = 2\vartheta_n \vartheta_{N-(n+1)m} \mathcal{D}^{1/2} \beta H(\varepsilon, n, m),$$

donde

$$\beta = \beta \left(\frac{(n+1)m-1}{2}, \frac{N-(n+1)m+1}{2} \right).$$

Usando ahora las cotas sobre $H(\varepsilon, n, m)$ obtenidas anteriormente se sigue:

$$\begin{aligned} 2\vartheta_n \vartheta_{N-(n+1)m} \mathcal{D}^{1/2} \beta \frac{2}{\vartheta_n} \frac{\vartheta_{m(n+1)-1}}{\sqrt{2\pi}} \left(\frac{(n+1)c\varepsilon}{n-m+2} \right)^{n-m+2} &\leq \\ &\leq I \leq \\ &\leq 2\vartheta_n \vartheta_{N-(n+1)m} \mathcal{D}^{1/2} \beta \frac{2}{\vartheta_n} \frac{\vartheta_{m(n+1)-1}}{\sqrt{2\pi}} \left(\frac{(n+1)^{3/2} C\varepsilon}{n-m+2} \right)^{n-m+2}. \end{aligned}$$

Usamos ahora la Desigualdad (4.1)

$$\begin{aligned} \frac{4\mathcal{D}^{1/2} \vartheta_N}{\sqrt{2\pi}} \left(\frac{2}{(n+1)m-1} \right)^{1/2} \left(\frac{(n+1)c\varepsilon}{n-m+2} \right)^{n-m+2} &\leq \\ &\leq I \leq \\ &\leq \frac{4\mathcal{D}^{1/2} \vartheta_N}{\sqrt{2\pi}} \left(\frac{N}{2} \right)^{1/2} \left(\frac{(n+1)^{3/2} C\varepsilon}{n-m+2} \right)^{n-m+2}, \end{aligned}$$

donde $0.245 \leq c \leq 2$ y $5.013 \leq C \leq 6.414$.

Concluimos la demostración recordando que

$$P = \mathbb{E}_\Delta [Vol_\varepsilon^m(f)] = \frac{I}{\vartheta_N},$$

luego

$$\frac{4}{\sqrt{\pi}} \left(\frac{\mathcal{D}}{(n+1)m-1} \right)^{1/2} \left(\frac{(n+1)c\varepsilon}{n-m+2} \right)^{n-m+2} \leq P \leq \frac{2}{\sqrt{\pi}} (N\mathcal{D})^{1/2} \left(\frac{(n+1)^{3/2}C\varepsilon}{n-m+2} \right)^{n-m+2}.$$

□

El siguiente corolario se sigue inmediatamente del Teorema 0.0.16.

Corolario 4.1.1. *Usando la notación anterior. Suponemos que $(d) = (d_1, \dots, d_m)$ son números naturales impares. Para todo número real ε con $\varepsilon \leq \frac{1}{n-m+2}$ se cumple:*

$$\begin{aligned} \frac{4}{\sqrt{\pi}} \frac{1}{\mathcal{D}^{1/2}\vartheta_{n-m}} \left(\frac{1}{(n+1)m-1} \right)^{1/2} \left(\frac{(n+1)c\varepsilon}{n-m+2} \right)^{n-m+2} &\leq \\ &\leq \mathbb{E}_\Delta [\mu_{norm}^{av}(f)] \leq \\ &\leq \frac{2}{\sqrt{\pi}} \frac{(N\mathcal{D})^{1/2}}{\vartheta_{n-m}} \left(\frac{(n+1)^{3/2}C\varepsilon}{n-m+2} \right)^{n-m+2}, \end{aligned}$$

donde $0.245 \leq c \leq 2$ y $5.013 \leq C \leq 6.414$.

Demostración. Trivial a partir de las cotas sobre $\nu_{n-m}[\mathbb{V}_{\mathbb{P}}(f)]$ dadas por el Corolario 1.3.5 y del Teorema 0.0.16. □

Nota 4.1.2. *Nótese que la cota inferior sigue siendo válida en el caso que la lista de grados $(d) = (d_1, \dots, d_n)$ contenga un número par.*

Antes de demostrar el Corolario 0.0.17 es necesario introducir el siguiente lema técnico que usaremos en la demostración.

Lema 4.1.3. *Sea X una variable aleatoria real y positiva tal que para todo número real positivo $t > 1$,*

$$\text{Prob}[X > t] \leq ct^{-\alpha},$$

donde $c > 1, \alpha > 1$ son dos constantes positivas. Se tiene entonces que:

$$\mathbb{E}[X] \leq c^{1/\alpha} \frac{\alpha}{\alpha - 1}.$$

Demostración. La demostración se puede encontrar en [Bel06]. □

4.2 Demostración del Corolario 0.0.17

Comenzamos por el primer ítem. A partir del Teorema 0.0.16 tenemos:

$$E_{\Delta} [Vol_{\varepsilon}^n(f)] = \frac{1}{\vartheta_N} \int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \sum_{\zeta \in V_{\mathbb{P}}(f)} \chi_{\varepsilon}(f, \zeta) dV_{\mathbb{P}}(f) d\nu_{\Delta} \leq \left(\frac{N\mathcal{D}}{\pi} \right)^{1/2} \frac{(n+1)^3 (C\varepsilon)^2}{2},$$

donde $\chi_{\varepsilon} : V^{\mathbb{R}} \rightarrow \{0, 1\}$ es la función característica del conjunto

$$A_{\varepsilon} = \{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) : \mu_{norm}(f, \zeta) \geq \varepsilon^{-1}\}.$$

Por otro lado:

$$\text{Prob}_{\Delta} [\mu_{worst}(f) \geq \varepsilon^{-1}] = \frac{1}{\vartheta_N} \int_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \chi'_{\varepsilon} d\nu_{\Delta},$$

donde $\chi'_{\varepsilon} : \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \rightarrow \{0, 1\}$ es la función característica del conjunto:

$$B_{\varepsilon} = \{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) : \mu_{worst}(f) \geq \varepsilon^{-1}\}.$$

La demostración se sigue teniendo en cuenta que:

$$\frac{1}{\vartheta_N} \int_{\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \chi'_{\varepsilon} d\nu_{\Delta} \leq \frac{1}{\vartheta_N} \int_{f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})} \sum_{\zeta \in V_{\mathbb{P}}(f)} \chi_{\varepsilon}(f, \zeta) dV_{\mathbb{P}}(f) d\nu_{\Delta}.$$

Para la segunda parte, simplemente aplicar el Teorema 4.1.3. □

Definimos ahora el condicionamiento μ medio, que denotaremos $\mu_{norm}^{av}(f)$, como el valor medio que alcanza el condicionamiento μ_{norm} en el conjunto de los ceros $V_{\mathbb{P}}(f)$. Es decir:

$$\mu_{norm}^{av}(f) = \frac{Vol_{\varepsilon}(f)}{\nu_{n-m}[V_{\mathbb{P}}(f)]} = \frac{1}{\nu_{n-m}[V_{\mathbb{P}}(f)]} \int_{V_{\mathbb{P}}(f)} \chi_{\varepsilon} dV_{\mathbb{P}}(f),$$

suponiendo que $\nu_{n-m}[V_{\mathbb{P}}(f)] > 0$ y denotando por $\chi_{\varepsilon} : V_{\mathbb{P}}(f) \rightarrow \{0, 1\}$ a la función característica del conjunto $A_{\varepsilon}(f) = \{\zeta \in V_{\mathbb{P}}(f) : \mu_{norm}(f, \zeta) > \varepsilon^{-1}\}$.

A partir de estas estimaciones se pueden construir procedimientos que calculen puntos en la variedad solución $V_{\mathbb{A}}(f)$ como se describe en [CKMW08].

El siguiente corolario se sigue del Teorema 0.0.16 (véase también la Proposición 0.0.15).

Corolario 4.2.1. *Usando la notación anterior. Para todo número real ε con $0 < \varepsilon < \min\{\frac{1}{3}, \frac{1}{n-m+2}\}$ se cumple*

$$E_{\Delta} [\nu_{n-m}[\zeta \in V_{\mathbb{P}}(f) : R(f, \zeta) < \varepsilon]] \leq 2 \left(\frac{N\mathcal{D}}{\pi} \right)^{1/2} \left(\frac{(n+1)^{3/2} C}{n-m+2} \right)^{n-m+2} \left(\frac{2u_0}{d^{3/2}} \varepsilon \right)^{n-m+2},$$

donde $5.013 \leq C \leq 6.414$ y $u_0 \approx 0.05992$ es una constante universal.

Demostración. Sea $\varepsilon < \min\{\frac{1}{3}, \frac{1}{n-m+2}\}$, entonces usando el Teorema 0.0.15 se tiene:

$$\min\left\{\frac{2u_0}{d^{3/2}\mu_{norm}(f, \zeta)}, \frac{1}{3}\right\} \leq R(f, \zeta).$$

Ahora, como

$$\nu_{n-m}[\zeta \in V_{\mathbb{P}}(f) : R(f, \zeta) < \varepsilon] \leq \nu_{n-m}[\zeta \in V_{\mathbb{P}}(f) : \frac{2u_0}{d^{3/2}\varepsilon} \leq \mu_{norm}(f, \zeta)],$$

usando las cotas del Corolario 0.0.16 concluimos:

$$E_{\Delta} [\nu_{n-m}[\zeta \in V_{\mathbb{P}}(f) : R(f, \zeta) < \varepsilon]] \leq 2 \left(\frac{N\mathcal{D}}{\pi}\right)^{1/2} \left(\frac{(n+1)^{3/2}C}{n-m+2}\right)^{n-m+2} \left(\frac{2u_0}{d^{3/2}\varepsilon}\right)^{n-m+2}.$$

□

4.3 Demostración del Corolario 0.0.18

A partir del Teorema 0.0.15 se puede obtener la siguiente desigualdad:

$$\min\left\{\frac{2u_0}{d^{3/2}\mu_{worst}(f)}, \frac{1}{3}\right\} \leq R_{worst}(f).$$

Por otro lado,

$$\min\{a, b\} \geq \frac{ab}{a+b}.$$

Juntando ambas expresiones obtenemos:

$$R_{worst}(f) \geq \frac{2u_0}{6u_0 + d^{3/2}\mu_{worst}(f)}.$$

Ahora, usando el Corolario 0.0.17 y el Teorema 1.3.7 concluimos

$$E_{\Delta} [R_{worst}] \geq \frac{2u_0\pi^{1/4}}{6u_0\pi^{1/4} + 2\sqrt{2}d^{3/2}(N\mathcal{D})^{1/4}(n+1)^{3/2}C}.$$

□

Capítulo 5

El número de componentes conexas

Καὶ τὰ τοῦ αὐτοῦ διπλάσια ἴσα ἀλλήλοις ἐστίν¹.
Ευκλείδης² [EucEC, Libro I, 5^o noción común], [Mor70]

A partir de la descripción de un algoritmo de deformación homotópica se concluye rápidamente que éstos sólo puede operar si el candidato inicial y el objetivo pertenecen a la misma componente conexa del espacio de sistema de ecuaciones polinomiales multivariadas a la que hemos sustraído la variedad discriminante. Es por ello que es muy interesante contar con una caracterización de las componentes conexas de la dicho conjunto.

Para cualquier espacio topológico X denotaremos por $\beta_0(X)$ el número de componentes conexas de X que en nuestros casos, este valor coincide con el 0-ésimo número de Betti.

Nuestros principales resultados al respecto son la caracterización de las componentes conexas de la variedad de incidencia sin el conjunto de soluciones singulares $\Sigma^{\mathbb{R}}$ (Teorema 0.0.27) así como una cota al número de componentes conexas del espacio de sistema de ecuaciones polinomiales multivariadas sin la variedad discriminante (Teorema 0.0.28).

5.1 Sistemas de ecuaciones lineales con coeficientes reales

Comenzamos estudiando las componentes conexas del espacio de sistemas de ecuaciones lineales. Se sabe que $\beta_0(\mathcal{O}(k, \mathbb{K})) = 2$ para todo $k \in \mathbb{N}$, $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$. De hecho contamos con una caracterización de $\mathcal{O}(k, \mathbb{K})$:

$$\mathcal{O}(k, \mathbb{K}) = \text{SO}(k, \mathbb{K}) \oplus \text{S}^- \mathcal{O}(k, \mathbb{K}),$$

donde

$$\text{SO}(k, \mathbb{K}) = \{U \in \mathcal{O}(k, \mathbb{K}) : \det(U) = 1\},$$

¹El todo es mayor que la parte.

²Euclides de Alejandría.

es el grupo ortogonal especial (también llamado *grupo de rotaciones*) y

$$S^- \mathcal{O}(k, \mathbb{K}) = \{U \in \mathcal{O}(k, \mathbb{K}) : \det(U) = -1\}.$$

Los siguientes resultados son bien conocidos y por lo tanto obviamos su demostración. Los incluimos en esta memoria con el fin de fijar las notaciones.

Lema 5.1.1. *Los siguientes subconjuntos semi-algebraicos de \mathbb{R}^m ,*

$$A_m := \{(x_1, \dots, x_m) \in \mathbb{R}^m : x_1 \geq x_2 \geq \dots \geq x_m \geq 0\},$$

$$A_m^+ := \{(x_1, \dots, x_m) \in \mathbb{R}^m : x_1 \geq x_2 \geq \dots \geq x_m > 0\},$$

son conexos y convexos:

Teorema 5.1.2 (Descomposición en Valores Singulares). *Sean $n \geq m$ dos números naturales, $x = (x_1, \dots, x_m)$ un vector en el conjunto A_m y sea $D(x) \in \mathcal{M}_{m \times n}(\mathbb{R})$ la matriz dada por*

$$D(x) = \left(\begin{array}{ccc|ccc} x_1 & \dots & 0 & 0 & \dots & 0 \\ & \ddots & & \vdots & & \vdots \\ 0 & \dots & x_m & 0 & \dots & 0 \end{array} \right).$$

Entonces la siguiente aplicación es continua y suprayectiva:

$$\begin{aligned} \Psi : \mathcal{O}(m, \mathbb{R}) \times A_m \times \mathcal{O}(n, \mathbb{R}) &\longrightarrow \mathcal{M}_{m \times n}(\mathbb{R}) \\ (U, x, W) &\longmapsto UD(x)W^t, \end{aligned}$$

donde $M^t \in \mathcal{M}_{n \times m}(\mathbb{R})$ denota la matriz transpuesta de M . Es más, si $S_{(m,n)} \subset \mathcal{M}_{m \times n}(\mathbb{R})$ denota la variedad algebraica de las matrices de rango menor o igual que $p - 1$, con $p = \min\{m, n\}$, la siguiente igualdad se cumple

$$\Psi(\mathcal{O}(m, \mathbb{R}) \times A_m^+ \times \mathcal{O}(n, \mathbb{R})) = \mathcal{M}_{m \times n}(\mathbb{R}) \setminus S_{(m,n)}.$$

El siguiente lema caracteriza las componentes conexas del conjunto de sistemas de ecuaciones lineales

Lema 5.1.3. *Usando la notación anterior.*

- *Si $n > m$, entonces $\beta_0(\mathcal{M}_{m \times n}(\mathbb{R}) \setminus S_{(m,n)}) = 1$.*
- *Si $m = n$, entonces $\beta_0(\mathcal{M}_n(\mathbb{R}) \setminus S_n) = 2$,*

donde $\mathcal{M}_n(\mathbb{R}) = \mathcal{M}_{n \times n}(\mathbb{R})$ y $S_n = S_{(n,n)}$.

Demostración. Como Ψ es una aplicación continua, los siguientes subconjuntos del espacio de matrices $\mathcal{M}_{m \times n}(\mathbb{R}) \setminus S_{(m,n)}$ son semi-algebraicos y conexos por caminos:

$$\begin{aligned} A_+ &:= \Psi(S\mathcal{O}(m, \mathbb{R}) \times A_m^+ \times S\mathcal{O}(n, \mathbb{R})). \\ B_{-, -} &:= \Psi(S^- \mathcal{O}(m, \mathbb{R}) \times A_m^+ \times S^- \mathcal{O}(n, \mathbb{R})). \\ A_- &:= \Psi(S^- \mathcal{O}(m, \mathbb{R}) \times A_m^+ \times S\mathcal{O}(n, \mathbb{R})). \\ B_{+, -} &:= \Psi(S\mathcal{O}(m, \mathbb{R}) \times A_m^+ \times S^- \mathcal{O}(n, \mathbb{R})). \end{aligned}$$

Se tiene entonces que:

$$\mathcal{M}_{m \times n}(\mathbb{R}) \setminus S_{(m,n)} = A_+ \cup B_{-,-} \cup A_- \cup B_{+,-}.$$

Afirmamos entonces que

$$A_+ = B_{-,-} \quad \text{y} \quad A_- = B_{+,-}. \quad (5.1)$$

Para probarlo, obsérvese que, para todo $x = (x_1, \dots, x_m) \in A_m^+$ y para todo par de matrices $U \in \mathcal{O}(m, \mathbb{R}), W \in \mathcal{O}(n, \mathbb{R})$ se cumple:

$$UD(x)W^t = UD(1^*)D(x)D(1^*)W^t, \quad (5.2)$$

donde $1^* = (-1, 1, \dots, 1)$. Denotamos ahora por $U_- \in \mathcal{O}(m, \mathbb{R})$ y $W_- \in \mathcal{O}(n, \mathbb{R})$ a las siguientes matrices

$$U_- = UD(1^*), \quad W_- = WD(1^*).$$

Se tiene entonces que:

- $U \in S\mathcal{O}(m, \mathbb{R})$ si y sólo si $U_- \in S^-\mathcal{O}(m, \mathbb{R})$.
- $W \in S\mathcal{O}(n, \mathbb{R})$ si y sólo si $W_- \in S^-\mathcal{O}(n, \mathbb{R})$.

En conclusión, la Igualdad (5.2) implica que

$$\mathcal{M}_{m \times n}(\mathbb{R}) \setminus S_{(m,n)} = A_+ \cup A_-$$

y por lo tanto:

$$\beta_0(\mathcal{M}_{m \times n}(\mathbb{R}) \setminus S_{(m,n)}) \leq 2.$$

Sea $n = m$, se tiene entonces que

$$\begin{aligned} A_+ &= \{M \in \mathcal{M}_n(\mathbb{R}) \setminus S_n : \det(M) > 0\}. \\ A_- &= \{M \in \mathcal{M}_n(\mathbb{R}) \setminus S_n : \det(M) < 0\}. \end{aligned}$$

En particular $A_+ \cap A_- = \emptyset$ y además ambos son conjuntos abiertos y cerrados en $\mathcal{M}_n(\mathbb{R}) \setminus S_n$. Luego son las dos componentes conexas de $\mathcal{M}_n(\mathbb{R}) \setminus S_n$ y por lo tanto:

$$\beta_0(\mathcal{M}_n(\mathbb{R}) \setminus S_n) = 2.$$

Por otro lado, sea $n > m$. En este caso se tiene que $A_+ \cap A_- \neq \emptyset$ y $\mathcal{M}_{m \times n}(\mathbb{R}) \setminus S_{(n,m)}$ es conexo. Obsérvese que:

$$(Id_m|0) = Id_m(Id_m|0)Id_n = Id_m(Id_m|0) \left(\begin{array}{c|c} Id_{n-1} & 0 \\ \hline 0 & -1 \end{array} \right).$$

Esto prueba que $(Id_m|0) \in A_+ \cap A_- \neq \emptyset$. □

Nota 5.1.4. En el caso que $n > m$ se tiene también que $A_+ = A_-$ debido a que si $H \in A_+$ entonces

$$H = UD(x)W,$$

con $U \in \mathcal{SO}(m, \mathbb{R})$, $W \in \mathcal{SO}(n, \mathbb{R})$, $x \in A_m^+$, se tiene también que

$$H = UD(x)W \left(\begin{array}{c|c} Id_{n-1} & 0 \\ \hline 0 & -1 \end{array} \right) = UD(x)W',$$

con $U \in \mathcal{SO}(m, \mathbb{R})$, $W' \in \mathcal{S}^-\mathcal{O}(n, \mathbb{R})$, $x \in A_m^+$ y $H \in A_-$. Luego

$$A_+ \subseteq A_-$$

La prueba del otro contenido $A_- \subseteq A_+$ es idéntica.

A partir de este lema, inmediatamente podemos conseguir resultados similares tanto para la esfera como para el espacio proyectivo del conjunto de sistemas lineales.

Corolario 5.1.5. Sea $\mathbb{S}^1(\mathcal{M}_n(\mathbb{R}))$ la esfera de radio 1 centrada en el origen del espacio de matrices reales cuadradas con respecto a la norma de Frobenius. Es decir:

$$\mathbb{S}^1(\mathcal{M}_n(\mathbb{R})) = \{M \in \mathcal{M}_n(\mathbb{R}) : \|M\|_F = 1\}.$$

Consideramos ahora el subconjunto abierto de $\mathbb{S}^1(\mathcal{M}_n(\mathbb{R}))$ dado por

$$\mathbb{S}^1(\text{GL}(n, \mathbb{R})) = \mathbb{S}^1(\mathcal{M}_n(\mathbb{R})) \setminus \left\{ \frac{M}{\|M\|_F} : M \in \mathcal{S}_n \setminus \{0\} \right\}.$$

Entonces, $\beta_0(\mathbb{S}^1(\text{GL}(n, \mathbb{R}))) = 2$ y las componentes conexas de $\mathbb{S}^1(\text{GL}(n, \mathbb{R}))$ son:

$$\begin{aligned} \mathbb{S}^1(A_+) &= \{M \in \mathbb{S}^1(\text{GL}(n, \mathbb{R})) : \det(M) > 0\}. \\ \mathbb{S}^1(A_-) &= \{M \in \mathbb{S}^1(\text{GL}(n, \mathbb{R})) : \det(M) < 0\}. \end{aligned}$$

Demostración. Consideramos la siguiente aplicación continua:

$$\begin{aligned} \pi : \mathcal{M}_n(\mathbb{R}) \setminus \{0\} &\longrightarrow \mathbb{S}^1(\mathcal{M}_n(\mathbb{R})) \\ M &\longmapsto \frac{M}{\|M\|_F}. \end{aligned}$$

Obviamente, se tiene que:

$$\mathbb{S}^1(A_+) = \pi(A_+), \quad \mathbb{S}^1(A_-) = \pi(A_-),$$

donde A_+ y A_- son los conjuntos definidos en la demostración del Lema 5.1.3. Tenemos entonces que

$$\beta_0(\mathbb{S}^1(\text{GL}(n, \mathbb{R}))) \leq 2.$$

Supongamos ahora que $U \in \mathbb{S}^1(A_+) \cap \mathbb{S}^1(A_-)$. Entonces, para todo $\lambda > 0$, $\lambda U \in A_+ \cap A_-$ que como vimos en la demostración del Lema 5.1.3 es un conjunto vacío. Hemos llegado a una contradicción, esto prueba que

$$\mathbb{S}^1(A_+) \cap \mathbb{S}^1(A_-) = \emptyset$$

y por lo tanto

$$\beta_0(\mathbb{S}^1(\text{GL}(n, \mathbb{R}))) = 2.$$

□

Corolario 5.1.6. Sea $\mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R}))$ el espacio proyectivo definido sobre el espacio vectorial de las matrices reales de tamaño $m \times n$. Denotamos por $\pi : \mathcal{M}_{m \times n}(\mathbb{R}) \setminus \{0\} \longrightarrow \mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R}))$ a la proyección canónica sobre el espacio proyectivo. Tenemos entonces que $\pi(S_{m \times n} \setminus \{0\})$ denota al conjunto de puntos proyectivos definidos por las matrices no singulares distintas de la matriz nula. Con estas notaciones se cumple

- Si $m < n$ entonces $\beta_0(\mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R})) \setminus \pi(S_{(m,n)} \setminus \{0\})) = 1$.
- Si $n = m$ y es impar entonces $\beta_0(\mathbb{P}(\mathcal{M}_n(\mathbb{R})) \setminus \pi(S_n \setminus \{0\})) = 1$.
- Si $n = m$ y es par entonces $\beta_0(\mathbb{P}(\mathcal{M}_n(\mathbb{R})) \setminus \pi(S_n \setminus \{0\})) = 2$.

Demostración. Sea $m < n$. Consideramos la aplicación continua dada por la proyección canónica

$$\pi : \mathcal{M}_{m \times n}(\mathbb{R}) \setminus \{0\} \longrightarrow \mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R})),$$

que envía a cada elemento a su clase. Usando el Lema 5.1.3 se tiene que

$$\pi(\mathbb{P}(\mathcal{M}_{m \times n}(\mathbb{R})) \setminus \pi(S_{(m,n)} \setminus \{0\})),$$

es conexo y el primer caso queda demostrado.

Sea ahora $n = m$. Tenemos entonces que $\pi(A_+)$ y $\pi(A_-)$ son conexos y que

$$\mathbb{P}(\mathcal{M}_n(\mathbb{R})) = \pi(A_+) \cup \pi(A_-),$$

por lo que

$$\beta_0(\mathbb{P}(\mathcal{M}_n(\mathbb{R}))) \leq 2.$$

Se nos presentan ahora los siguientes casos

- Si $n = m$ es par, $\pi(A_+) \cap \pi(A_-) = \emptyset$ y por lo tanto la tercera afirmación se cumple.
- Si $n = m$ es impar, afirmamos que $\pi(A_+) = \pi(A_-) = \mathbb{P}(\mathcal{M}_n(\mathbb{R}))$ y por lo tanto se sigue el segundo ítem.

Probemos dicha afirmación. Para ello escogemos una matriz $X \in \mathbb{P}(\mathcal{M}_n(\mathbb{R})) \setminus \pi(S_n \setminus \{0\})$ entonces existe una matriz $M \in \mathcal{M}_n(\mathbb{R})$ tal que para todo $\lambda \in \mathbb{R} \setminus \{0\}$ se cumple $\pi(\lambda M) = X$.

Supongamos que $M \in A_+$ y tomemos dos números reales $\lambda_- < 0 < \lambda_+$ tenemos entonces que:

$$\lambda_+ M \in A_+ \text{ y } \lambda_- M \in A_-,$$

donde A_+ y A_- son los conjuntos definidos en la demostración del Lema 5.1.3 anterior. En el caso que $M \in A_-$ se tiene que:

$$\lambda_+ M \in A_- \text{ y } \lambda_- M \in A_+.$$

En cualquiera de los casos se tiene que

$$\pi(\lambda_+ M) = \pi(\lambda_- M),$$

y por lo tanto concluimos que $X \in \pi(A_+) \cap \pi(A_-)$ y se sigue la afirmación. \square

5.2 Sistemas de ecuaciones polinomiales con coeficientes reales

Tratamos ahora de resolver la caracterización de las componentes conexas en el caso no lineal. Es decir, en el espacio de los sistemas de ecuaciones polinomiales homogéneas multivariadas con coeficientes reales. En efecto, esta sección está dedicada a demostrar el Teorema 0.0.27.

Para ello necesitamos introducir algunas notaciones. Sea $\widetilde{\Sigma}^{\mathbb{R}}$ el subconjunto de $\mathcal{H}_{(d)}^{\mathbb{R}} \times \mathbb{P}_n(\mathbb{R})$ dado por la siguiente igualdad:

$$\widetilde{\Sigma}^{\mathbb{R}} := (\pi \times Id_{\mathbb{P}})^{-1}(\Sigma^{\mathbb{R}}),$$

donde $Id_{\mathbb{P}} : \mathbb{P}_n(\mathbb{R}) \rightarrow \mathbb{P}_n(\mathbb{R})$ denota la aplicación identidad, $\pi : \mathcal{H}_{(d)}^{\mathbb{R}} \rightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ la proyección canónica y $\pi \times Id_{\mathbb{P}}$ el producto cartesiano de ambas aplicaciones.

Lema 5.2.1. *Usando la notación anterior. Para todo $x \in \mathbb{P}_n(\mathbb{R})$ se cumple:*

- Si $m < n$ entonces $\beta_0(\widetilde{V}_x^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}}) = 1$.
- Si $m = n$ entonces $\beta_0(\widetilde{V}_x^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}}) = 2$.

Demostración. Como en anteriores ocasiones sólo detallaremos el caso $x = e_0$. Nótese que para todo $f \in \widetilde{V}_{e_0}^{\mathbb{R}}$, f se puede descomponer de forma única como la suma $f = f_v + f_l$ donde $f_v \in V'_{e_0}$ y $f_l \in L_{e_0}$. Trivialmente se obtiene que:

$$D_{e_0}f = D_{e_0}f_v + D_{e_0}f_l = D_{e_0}f_l.$$

Y por lo tanto:

$$T_{e_0}f = D_{e_0}f_l|_{\langle e_0 \rangle^\perp} = T_{e_0}f_l.$$

Es más, la siguiente aplicación:

$$\begin{aligned} \tilde{\varphi} : V'_{e_0} \times \mathcal{M}_{m \times n}(\mathbb{R}) &\longrightarrow \widetilde{V}_{e_0}^{\mathbb{R}} = V'_{e_0} \perp L_{e_0} \\ (h, M) &\longmapsto h + \varphi(M), \end{aligned}$$

es biyectiva y una isometría por lo que es también un homeomorfismo. Finalmente obsérvese que:

$$\tilde{\varphi}(V'_{e_0} \times (\mathcal{M}_{m \times n}(\mathbb{R}) \setminus S_{(m,n)})) = \widetilde{V}_{e_0}^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}}.$$

Ahora, usando el Lema 5.1.3 y suponiendo que $m < n$, $\widetilde{V}_{e_0}^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}}$ es conexo mientras que si $m = n$ entonces $\beta_0(\widetilde{V}_{e_0}^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}}) = 2$. Es más, en el caso que $m = n$, las componentes conexas de $\widetilde{V}_{e_0}^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}}$ están dadas por las siguientes identidades:

$$\widetilde{V}_{e_0}^{\mathbb{R}+} = \tilde{\varphi}(V'_{e_0} \times A_+), \quad \widetilde{V}_{e_0}^{\mathbb{R}-} = \tilde{\varphi}(V'_{e_0} \times A_-),$$

donde A_+ y A_- son las dos componentes conexas de $\mathcal{M}_{m \times n}(\mathbb{R}) \setminus S_{(m,n)}$ definidas en la demostración del Lema 5.1.3. \square

Nota 5.2.2. Obsérvese que para todo $x \in \mathbb{S}^1(\mathbb{R}^n)$ y para toda aplicación polinomial homogénea $f \in \mathcal{H}_{(d)}^{\mathbb{R}}$ tal que $f(x) = 0$, el gradiente de f en x , $\nabla_x f$, es ortogonal a x . Es decir:

$$\langle \nabla_x f, x \rangle = 0, \quad \forall x \in V_{\mathbb{A}}^{\mathbb{R}}(f).$$

Tomamos ahora $(f, x) \in V^{\mathbb{R}}$ tal que $x \in V_{\mathbb{A}}^{\mathbb{R}}(f)$. Entonces las filas de la matriz jacobiana

$$D_x f = \begin{pmatrix} \frac{\partial f_1}{\partial x_0} \Big|_x & \cdots & \frac{\partial f_1}{\partial x_n} \Big|_x \\ \vdots & & \vdots \\ \frac{\partial f_m}{\partial x_0} \Big|_x & \cdots & \frac{\partial f_m}{\partial x_n} \Big|_x \end{pmatrix},$$

son todas ortogonales a x . En particular, si $(f, x) \notin \Sigma'^{\mathbb{R}}$ la siguiente matriz tiene rango máximo:

$$\Delta_x f = \begin{pmatrix} x^t \\ D_x f \end{pmatrix} \in \mathcal{M}_{(m+1) \times (n+1)}(\mathbb{R}) \setminus S_{(m+1, n+1)}.$$

En el caso que $n = m$ inmediatamente se tiene que las filas de $D_x f$ definen una base del espacio tangente $T_x \mathbb{P}_n(\mathbb{R}) = \langle x \rangle^{\perp}$ y $\Delta_x f$ es una matriz regular. Es decir $\Delta_x f \in \text{GL}(n+1, \mathbb{R})$.

Nota 5.2.3. Sea $\tilde{e}_0 = (1, 0, \dots, 0) \in \mathbb{R}^{n+1}$ un representante del punto proyectivo $e_0 = (1 : 0 : \dots : 0) \in \mathbb{P}_n(\mathbb{R})$. Para todo $f \in \widetilde{V}_{e_0}^{\mathbb{R}}$ definimos el jacobiano extendido de f en e_0 como la siguiente matriz.

$$\Delta_{e_0} f := \begin{pmatrix} \tilde{e}_0^t \\ D_{\tilde{e}_0} f \end{pmatrix}.$$

Es más, sea $(f, e_0) \in \widetilde{V}_{e_0}^{\mathbb{R}}$, entonces, debido a la Nota 5.2.2, se tiene que para todo número natural j tal que $1 \leq j \leq m$ se cumple:

$$\langle \nabla_{e_0} f_j, e_0 \rangle = \frac{\partial f_j}{\partial x_0} \Big|_{e_0} = 0.$$

Y también

$$D_{e_0} f(e_0) = 0,$$

Luego, la primera columna de $D_{e_0} f$ es nula. En particular se tiene que:

$$\Delta_{e_0} f = \left(\begin{array}{c|ccc} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & D_{\tilde{e}_0} f(e_1) & \cdots & D_{\tilde{e}_0} f(e_n) \\ 0 & & & \end{array} \right),$$

donde $D_{e_0} f(e_i)$ denota la columna que resulta de multiplicar $D_{e_0} f$ por el i -ésimo vector de la base canónica $\{e_0, \dots, e_n\}$ de \mathbb{R}^{n+1}

Pasamos ahora a introducir el el eje central de nuestro discurso.

Definición 5.2.4 (Jacobiano extendido orientado). *Dado un par $(f, x) \in V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$ y $U \in \text{SO}(n+1, \mathbb{R})$ una rotación tal que $Ue_0 = x$. Si $\tilde{x} \in \mathbb{R}^{n+1}$ es el punto afín tal que $\tilde{x} = U\tilde{e}_0$, entonces, definimos el jacobiano extendido orientado de f en x por la siguiente matriz*

$$\Delta_x f := \begin{pmatrix} \tilde{x}^t \\ D_{\tilde{x}} f \end{pmatrix}.$$

Nótese que en el caso que $m = n$, el signo del $\det(\Delta_x f)$ es independiente de la rotación $U \in \text{SO}(n+1, \mathbb{R})$ escogida.

Corolario 5.2.5. *Usando la notación anterior. Si $m = n$ entonces las dos componentes conexas de $\widetilde{V}_x^{\mathbb{R}} \setminus \widetilde{\Sigma}'^{\mathbb{R}}$ están dadas por las siguientes igualdades:*

$$\widetilde{V}_x^{\mathbb{R}+} = \{(f, x) \in \widetilde{V}_x^{\mathbb{R}} : \det(\Delta_x f) > 0\}.$$

$$\widetilde{V}_x^{\mathbb{R}-} = \{(f, x) \in \widetilde{V}_x^{\mathbb{R}} : \det(\Delta_x f) < 0\}.$$

Corolario 5.2.6. *Sea $\mathbb{S}(\widetilde{V}_x^{\mathbb{R}})$ la esfera en el espacio vectorial $\widetilde{V}_x^{\mathbb{R}}$ con respecto del producto escalar de Bombieri $\langle \cdot, \cdot \rangle_{\Delta}$. Es decir:*

$$\mathbb{S}(\widetilde{V}_x^{\mathbb{R}}) : \{(f, x) \in \widetilde{V}_x^{\mathbb{R}} : \|f\|_{\Delta} = 1\}.$$

Se tiene que

- *Si $m < n$ entonces $\beta_0(\mathbb{S}(\widetilde{V}_x^{\mathbb{R}}) \setminus \widetilde{\Sigma}'^{\mathbb{R}}) = 1$.*
- *Si $m = n$ entonces $\beta_0(\mathbb{S}(\widetilde{V}_x^{\mathbb{R}}) \setminus \widetilde{\Sigma}'^{\mathbb{R}}) = 2$.*

Corolario 5.2.7. *Usando la notación anterior.*

- *Si $m < n$ entonces $\beta_0(V_x^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}) = 1$.*
- *Si $m = n$ y n es impar, entonces $\beta_0(V_x^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}) = 1$.*
- *Si $m = n$ y n es par, entonces $\beta_0(V_x^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}) = 2$.*

Teorema 5.2.8. *Sea $\widetilde{V}^{\mathbb{R}} \subseteq \mathcal{H}_{(d)}^{\mathbb{R}} \times \mathbb{P}_n(\mathbb{R})$ el siguiente conjunto:*

$$\widetilde{V}^{\mathbb{R}} := \{(f, x) \in \mathcal{H}_{(d)}^{\mathbb{R}} \times \mathbb{P}_n(\mathbb{R}) : f(x) = 0 \in \mathbb{R}^m\}.$$

A partir de esta definición se tiene que:

- *Si $m < n$ entonces $\beta_0(\widetilde{V}^{\mathbb{R}} \setminus \widetilde{\Sigma}'^{\mathbb{R}}) = 1$.*
- *Si $m = n$ entonces $\beta_0(\widetilde{V}^{\mathbb{R}} \setminus \widetilde{\Sigma}'^{\mathbb{R}}) = 2$.*

Es más, en este último caso las componentes conexas son:

$$\widetilde{V}^{\mathbb{R}^+} = \left\{ (f, x) \in \widetilde{V}^{\mathbb{R}} : \det(\Delta_x f) > 0 \right\}.$$

$$\widetilde{V}^{\mathbb{R}^-} = \left\{ (f, x) \in \widetilde{V}^{\mathbb{R}} : \det(\Delta_x f) < 0 \right\}.$$

Demostración. Para empezar extenderemos a $\widetilde{V}_{e_0}^{\mathbb{R}}$ la aplicación definida en (1.1). Es decir, definimos la siguiente aplicación continua:

$$\begin{aligned} \widetilde{\Phi} : \widetilde{V}_{e_0}^{\mathbb{R}} \times \mathcal{SO}(n+1, \mathbb{R}) &\longrightarrow \widetilde{V}^{\mathbb{R}} \\ ((f, e_0), U) &\longmapsto (f \circ U^{-1}, Ue_0), \end{aligned}$$

En el caso que $m < n$, como $\mathcal{SO}(n+1, \mathbb{R})$ es conexo, a partir del Lema 5.2.1 y del Corolario 1.2.6 se sigue que $\widetilde{V}^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}}$ es conexo.

Para el caso $m = n$, el mismo Lema 5.2.1 y el mismo Corolario 1.2.6 implican que podemos descomponer $\widetilde{V}^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}}$ en dos conjuntos semi-algebraicos. Es decir:

$$\widetilde{V}^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}} = \widetilde{V}^{\mathbb{R}^+} \cup \widetilde{V}^{\mathbb{R}^-},$$

donde $\widetilde{V}^{\mathbb{R}^+}$, $\widetilde{V}^{\mathbb{R}^-}$ son los conjuntos definidos por las siguientes igualdades:

$$\begin{aligned} \widetilde{V}^{\mathbb{R}^+} &= \widetilde{\Phi}(\widetilde{V}_{e_0}^{\mathbb{R}^+} \times \mathcal{SO}(n+1, \mathbb{R})) \\ \widetilde{V}^{\mathbb{R}^-} &= \widetilde{\Phi}(\widetilde{V}_{e_0}^{\mathbb{R}^-} \times \mathcal{S}^-\mathcal{O}(n+1, \mathbb{R})). \end{aligned}$$

Por otro lado tenemos que a partir del Corolario 1 del Capítulo 12 de [BCSS98], se sigue que para toda matriz $U \in \mathcal{SO}(n+1, \mathbb{R})$ la siguiente aplicación es una isometría:

$$\begin{aligned} \widetilde{\Phi}_U : \widetilde{V}_{e_0}^{\mathbb{R}} &\longrightarrow \widetilde{V}^{\mathbb{R}} \\ (f, e_0) &\longmapsto (f \circ U^{-1}, Ue_0), \end{aligned}$$

Luego, $\widetilde{V}^{\mathbb{R}^+}$ y $\widetilde{V}^{\mathbb{R}^-}$ son dos subconjuntos abiertos y conexos de $\widetilde{V}^{\mathbb{R}}$. Es más, se tiene que:

$$\widetilde{V}^{\mathbb{R}^+} = \left\{ (f, x) \in \widetilde{V}^{\mathbb{R}} : \det(\Delta_x f) > 0 \right\}$$

$$\widetilde{V}^{\mathbb{R}^-} = \left\{ (f, x) \in \widetilde{V}^{\mathbb{R}} : \det(\Delta_x f) < 0 \right\}.$$

En particular, son conjuntos disjuntos y por lo tanto las dos componentes conexas de $\widetilde{V}^{\mathbb{R}} \setminus \widetilde{\Sigma}^{\mathbb{R}}$. \square

5.2.1 Demostración del Teorema 0.0.27

En el caso que $m < n$, como $S\mathcal{O}(n+1, \mathbb{R})$ es una componente conexa, el Lema 5.2.7 y el Corolario 1.2.6 implican que $V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$ es una componente conexa y por lo tanto se sigue la primera afirmación.

En el caso que $m = n$ con n impar, $V_{e_0}^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$ es una componente conexa y por lo tanto

$$\Phi((V_{e_0}^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}) \times S\mathcal{O}(n+1, \mathbb{R})) = V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$$

también lo es. Se sigue por tanto la segunda afirmación.

Finalmente, si $m = n$ con n impar, el mismo Lema 5.2.7 y el mismo Corolario 1.2.6 implican que podemos descomponer $V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$ en dos conjuntos semi-algebraicos. Es decir:

$$V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}} = V^{\mathbb{R}^+} \cup V^{\mathbb{R}^-},$$

donde $V^{\mathbb{R}^+}$, $V^{\mathbb{R}^-}$ son los siguientes conjuntos

$$\begin{aligned} V^{\mathbb{R}^+} &= \Phi(V_{e_0}^{\mathbb{R}^+} \times S\mathcal{O}(n+1, \mathbb{R})) \\ V^{\mathbb{R}^-} &= \Phi(V_{e_0}^{\mathbb{R}^-} \times S^-\mathcal{O}(n+1, \mathbb{R})). \end{aligned}$$

A partir del Corolario 1 del Capítulo 12 de [BCSS98], se sigue que para toda matriz $U \in S\mathcal{O}(n+1, \mathbb{R})$ la siguiente aplicación es una isometría:

$$\begin{aligned} \Phi_U : \quad V_{e_0}^{\mathbb{R}} &\longrightarrow V^{\mathbb{R}} \\ (f, e_0) &\mapsto (f \circ U^{-1}, Ue_0), \end{aligned}$$

luego $V^{\mathbb{R}^+}$ y $V^{\mathbb{R}^-}$ son dos subconjuntos abiertos y conexos de $V^{\mathbb{R}}$. Es más, se tiene que:

$$\begin{aligned} V^{\mathbb{R}^+} &= \{(f, x) \in V^{\mathbb{R}} : \det(\Delta_x f) > 0\} \\ V^{\mathbb{R}^-} &= \{(f, x) \in V^{\mathbb{R}} : \det(\Delta_x f) < 0\}. \end{aligned}$$

En particular, son conjuntos disjuntos y por lo tanto las dos componentes conexas de $V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$. Se cumple entonces la tercera afirmación. \square

5.3 Sistemas de ecuaciones polinomiales con coeficientes reales y soluciones complejas

Tratamos ahora el caso no lineal cuando consideramos las soluciones reales y complejas. Es decir, estudiamos el número de componentes conexas del espacio $V^{\mathbb{R}, \mathbb{C}}$. Con el objetivo de simplificar la notación a partir de ahora denotaremos por $\pi_1 = \pi_1^{\mathbb{R}, \mathbb{C}}$ y $\pi_2 = \pi_2^{\mathbb{R}, \mathbb{C}}$. Además, denotaremos por $\Sigma = \Sigma^{\mathbb{R}, \mathbb{C}}$ y $\Sigma' = \Sigma'^{\mathbb{R}, \mathbb{C}}$.

Como es habitual, consideramos la siguiente acción del grupo ortogonal especial sobre la variedad de incidencia

$$\begin{aligned} V^{\mathbb{R}, \mathbb{C}} \times S\mathcal{O}(n+1, \mathbb{R}) &\longrightarrow V^{\mathbb{R}, \mathbb{C}} \\ ((f, x), U) &\mapsto (f \circ U^{-1}, Ux). \end{aligned}$$

Enunciamos ahora dos lemas técnicos cuya demostración es inmediata.

Lema 5.3.1. *Dado un par $(f, x) \in V^{\mathbb{R}, \mathbb{C}}$, y una matriz $U \in \text{SO}(n+1, \mathbb{R})$, entonces $Ux \in \mathbb{P}_n(\mathbb{C})$ si y solo si $x \in \mathbb{P}_n(\mathbb{C})$. En particular, para toda matriz $U \in \text{SO}(n+1, \mathbb{R})$ y para todo sistema de ecuaciones $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ el número de soluciones reales (i.e. soluciones en $\mathbb{P}_n(\mathbb{R})$) de f y $f \circ U^{-1}$ coincide.*

Lema 5.3.2. *La acción del grupo de rotaciones preserva la variedad singular en todos los casos. Es decir,*

- Para toda matriz $U \in \text{SO}(n+1, \mathbb{K})$ y para todo par $(f, x) \in \Sigma'^{\mathbb{K}}$ se tiene que:

$$(f \circ U^{-1}, Ux) \in \Sigma'^{\mathbb{K}}.$$

- Para toda matriz $U \in \text{SO}(n+1, \mathbb{K})$ y para todo par $f \in \Sigma^{\mathbb{K}}$ se tiene que:

$$(f \circ U^{-1}) \in \Sigma^{\mathbb{K}}.$$

Como consecuencia inmediata se tiene que:

Corolario 5.3.3. *Las componentes conexas de $V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma'$ son invariantes respecto de la acción del grupo $\text{SO}(n+1, \mathbb{R})$. En particular, para cada componente conexa C de $V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma'$ que interseque $V^{\mathbb{R}}$ entonces también interseca a $V_{e_0}^{\mathbb{R}, \mathbb{C}}$, donde $e_0 = (1 : 0 : \dots : 0) \in \mathbb{P}_n(\mathbb{R})$. Es más, se tiene que:*

$$\beta_0 \left(V_{e_0}^{\mathbb{R}, \mathbb{C}} \setminus \Sigma' \right) \geq \# \left\{ C \subseteq V^{\mathbb{R}, \mathbb{C}} : \begin{array}{l} C \text{ es una componente conexa por caminos} \\ V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma' \text{ y } C \cap V^{\mathbb{R}} \neq \emptyset \end{array} \right\}.$$

Demostración. Para la primera afirmación, ya que todas las órbitas son caminos pertenecerán a la misma componente conexa de $V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma'$. Además, por cada componente conexa por caminos C de $V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma'$ que interseque a $V^{\mathbb{R}}$ existe un par $(f, x) \in C \cap V^{\mathbb{R}}$ y por lo tanto un rotación $U \in \text{SO}(n+1, \mathbb{R})$ tal que $Ux = e_0$. Luego, $(f \circ U^{-1}, e_0)$ está en la órbita de (f, x) y por lo tanto $(f \circ U^{-1}, e_0) \in C$.

Para la segunda afirmación, como toda componente conexa por caminos C de $V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma'$ tal que $C \cap V^{\mathbb{R}} \neq \emptyset$ cumple que $C \cap V_{e_0}^{\mathbb{R}, \mathbb{C}} \neq \emptyset$ inmediatamente podemos concluir que:

$$\beta_0 \left(V_{e_0}^{\mathbb{R}, \mathbb{C}} \setminus \Sigma' \right) \geq \beta_0 \left(V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma' \right).$$

Esto es debido a que el conjunto $C \cap (V_{e_0}^{\mathbb{R}, \mathbb{C}} \setminus \Sigma')$ es abierto y cerrado en $V_{e_0}^{\mathbb{R}, \mathbb{C}} \setminus \Sigma'$ y por lo tanto se tiene la desigualdad. \square

Corolario 5.3.4. *Las componentes conexas de $V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$ también son invariantes respecto de la acción del grupo $\text{SO}(n+1, \mathbb{R})$. Además, para toda componente conexa C de $V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$ se tiene que: $C \cap V_{e_0}^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}}$. Es más, se tiene que:*

$$\beta_0 \left(V_{e_0}^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}} \right) \geq \beta_0 \left(V^{\mathbb{R}} \setminus \Sigma'^{\mathbb{R}} \right).$$

Demostración. Ambas afirmaciones se siguen inmediatamente de los lemas previos. \square

Una proposición en el mismo sentido es:

Proposición 5.3.5. *Todas las componentes conexas de $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma$ son invariantes respecto de la acción del grupo $\mathrm{SO}(n+1, \mathbb{R})$. Además, en el caso que $n = m$, si dos sistemas $f, g \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma$ pertenecen a la misma componente conexa de $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma$ entonces tienen el mismo número de soluciones reales.*

Demostración. La primera afirmación se sigue por los mismos argumentos que los anteriores teoremas.

Para la segunda afirmación, supongamos que f, g pertenecen a la misma componente conexa por caminos de $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma$. Entonces, podemos construir un camino $\gamma : [0, 1] \rightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma$ tal que $\gamma(0) = f$ y $\gamma(1) = g$ (cf. [BCR87] para los detalles). Es más, como $\Gamma = \gamma([0, 1])$ es compacto, existe un entorno cerrado tubular Γ_ϵ of Γ tal que $\Gamma_\epsilon \cap \Sigma = \emptyset$ y Γ_ϵ esta contenido en la misma componente conexa por caminos de $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma$ que contiene a f y a g . Además, a partir del Teorema del Número de Condicionamiento (cf. [SS93a]) podemos suponer que existe una constante real $H > 0$ tal que $\mu_{worst}(h) \leq H$ para todos los sistemas $h \in \Gamma_\epsilon$. Pero, es más, existe un camino *regular a trozos* C contenido en Γ_ϵ tal que une f con g . Entendemos por *regular a trozos* un conjunto finito de curvas $\{\gamma_i : [0, 1] \rightarrow \Gamma_\epsilon\}_{1 \leq i \leq r}$ con las siguientes propiedades:

- $\bigcup_{i=1}^r \gamma_i([0, 1]) \subseteq \Gamma_\epsilon$.
- $\gamma_1(0) = f, \gamma_r(1) = g$.
- $\gamma_i(1) = \gamma_{i+1}(0)$ para todo $1 \leq i \leq r - 1$.

Usando inducción en r probaremos que el número de soluciones en $\mathbb{P}_n(\mathbb{R})$ del sistema $\gamma_1(0)$ es el mismo que el del sistema $\gamma_r(1)$. Usaremos la Proposición 3 del Capítulo 14 de [BCSS98] (o su formulación original en [SS93a]).

Primero observamos que si una raíz $z \in \mathbb{P}_n(\mathbb{C})$ de $f \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ tiene como cero aproximado asociado $\tilde{z} \in \mathbb{P}_n(\mathbb{R})$ entonces z también pertenece a $\mathbb{P}_n(\mathbb{R})$. Esto es debido a que $\mathbb{P}_n(\mathbb{R})$ es un conjunto cerrado en $\mathbb{P}_n(\mathbb{C})$.

Probaremos el caso $r = 1$ y dejamos el paso de inducción para el lector. Como hemos visto con anterioridad $\pi_1 : V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma \rightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ es una aplicación recubridora. Ahora, si $f = \gamma_1(0)$ y $\{z_1, \dots, z_R\}$ son las R soluciones reales distintas de f , es decir $V_{\mathbb{P}}^{\mathbb{R}}(f) = \{z_1, \dots, z_R\}$ y $\sharp V_{\mathbb{P}}^{\mathbb{R}}(f) = R$ se tiene que existen R arcos diferenciables $\gamma'_i : [0, 1] \rightarrow V^{\mathbb{R}, \mathbb{C}} \setminus \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$, tales que:

- $\gamma'_i(0) = (f, z_i), 1 \leq i \leq R$.
- $\pi_1 \circ \gamma'_i(t) = \gamma_i(t)$ para todo $t \in [0, 1]$.
- $\pi_1(\gamma'_i(1)) = g \forall i$.
- $\gamma'_i([0, 1]) \cap \gamma'_j([0, 1]) = \emptyset \forall i \neq j$.

Usando la Proposición 3 del Capítulo 12 de [BCSS98], para todo $i, 1 \leq i \leq R$, existe una partición del intervalo $[0, 1]$

$$0 = s_0 < \dots < s_k = 1,$$

tal que la secuencia

$$\begin{aligned} z_0 &= z_i \\ z_{j+1} &= N_{f_{s_{j+1}}}(z_j), \quad 0 \leq j \leq k-1, \end{aligned}$$

donde $f_{s_j} = \pi_1^{\mathbb{R}}(\gamma'_i(s_j))$, satisface

$$z_j \text{ es un cero aproximado de } f_{s_j} \text{ con cero asociado } \pi_2^{\mathbb{R}}(\gamma'_i(s_j)).$$

Como $f_{s_j} \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$ para todo j y $z_i \in \mathbb{P}_n(\mathbb{R})$ concluimos que:

- $z_k \in \mathbb{P}_n(\mathbb{R})$.
- z_k es un cero aproximado de un cero de $V_{\mathbb{P}}^{\mathbb{R}}(g)$.

Luego, como $g \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma$ concluimos inmediatamente que $\sharp V_{\mathbb{P}}^{\mathbb{R}}(g) \geq \sharp V_{\mathbb{P}}^{\mathbb{R}}(f)$. La otra desigualdad se sigue simplemente intercambiando f y g . Podemos concluir entonces que $\sharp V_{\mathbb{P}}^{\mathbb{R}}(g) = \sharp V_{\mathbb{P}}^{\mathbb{R}}(f)$ y el caso $r = 1$ se sigue. La hipótesis de inducción se sigue por el mismo argumento. \square

5.3.1 Demostración del Teorema 0.0.28

La primera afirmación es una consecuencia directa de la Proposición 5.3.5 anterior. Para la segunda afirmación, como hemos visto también en la Proposición 5.3.5,

$$\pi_1 : V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma' \longrightarrow \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma$$

es una aplicación recubridora, luego π_1 es una aplicación suprayectiva y por lo tanto

$$\beta_0(\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma) \leq \beta_0(V^{\mathbb{R}, \mathbb{C}} \setminus \Sigma')$$

y se sigue la segunda afirmación. \square

Capítulo 6

Camino en la variedad de incidencia

*δεν υπάρχει βασιλικός δρόμος για τη γεωμετρία*¹.
Atribuido a **Ευκλείδης**² [Pro50], [Mor70]

Como ya habíamos comentado anteriormente, uno de los procesos claves en la construcción de un algoritmo de deformación homotópica es poder construir caminos en la variedad de incidencia. Ahora bien, a tenor de lo visto en el Capítulo 5 nos encontramos con diversos problemas:

- Si $m = n$ es un número par, entonces la variedad de incidencia tiene dos componentes conexas.
- Las componentes conexas de la variedad de incidencia no son convexas.

Por todo ello es necesario replantearse la forma de escoger los puntos iniciales del algoritmo de deformación homotópica pues primero es necesario conseguir un punto que se encuentre en la misma componente conexa y luego habrá que describir un procedimiento adecuado que garantice no solo que existe el camino, sino que el número de pasos no sea excesivo. A continuación presentamos una posible idea que en su estado actual no puede ser utilizada para la creación de un algoritmo de deformación homotópica pues requiere tener información sobre la raíz que no conocemos.

Comenzaremos introduciendo las curvas isobáricas en el grupo ortogonal especial $SO(n, \mathbb{R})$. Los lemas son conocidos pero los usaremos para fijar la notación.

Lema 6.0.6. *Para toda matriz $U \in SO(n, \mathbb{R})$, existe otra matriz $L \in \mathcal{M}_n(\mathbb{R})$ tal que $iL \in \mathcal{M}_n(\mathbb{C})$ es una matriz antisimétrica y la siguiente aplicación*

$$\begin{aligned} \gamma : [0, 1] &\longrightarrow \mathcal{M}_n(\mathbb{C}) \\ t &\longmapsto e^{iLt}, \end{aligned}$$

cumple:

⁰Euclides de Alejandría.

¹Majestad, no hay camino *real* hacia la geometría. Nótese que *real* aquí se refiere a la realeza. Quizás la traducción inglesa sea más clarificadora: «Sire, there is no royal road to geometry».

²Euclides de Alejandría.

- γ es continua y diferenciable.
- $\gamma([0, 1]) \subseteq \text{SO}(n, \mathbb{C})$.
- $\gamma(0) = Id_n, \gamma(1) = U$.

Demostración. Comenzamos con algunas notaciones elementales que describen la aplicación γ .

Dado un número $r \in \mathbb{N}$ tal que $1 \leq r \leq n$, y dadas dos matrices $A \in \mathcal{M}_r(\mathbb{C})$, $B \in \mathcal{M}_{n-r}(\mathbb{C})$ denotamos por $A \oplus B$ a la suma directa de A y B . Es decir,

$$A \oplus B = \begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix} \in \mathcal{M}_n(\mathbb{C}).$$

Ahora, para toda $\theta \in (0, \pi]$ denotaremos por $\mathcal{R}(\theta) \in \mathcal{M}_2(\mathbb{R})$ a la matriz que codifica la *rotación plana* de ángulo θ . Es decir:

$$\mathcal{R}(\theta) = \begin{pmatrix} \cos \theta & + \sin \theta \\ - \sin \theta & \cos \theta \end{pmatrix}.$$

Denotaremos la *Forma Canónica de Jordan* de la matriz $\mathcal{R}(\theta)$ por $\mathcal{J}(\theta) \in \mathcal{M}_2(\mathbb{C})$. Es decir:

$$\mathcal{J}(\theta) = \begin{pmatrix} \cos \theta + i \sin \theta & 0 \\ 0 & \cos \theta - i \sin \theta \end{pmatrix}.$$

Además, denotaremos por $M_\theta \in \text{GL}(2, \mathbb{C})$ a las matrices regulares tales que

$$M_\theta \mathcal{R}(\theta) M_\theta^{-1} = \mathcal{J}(\theta).$$

Sea ahora $U \in \text{SO}(n, \mathbb{R})$ una rotación, entonces el *Teorema de Descomposición Espectral* nos asegura que existe una matriz $V \in \mathcal{O}(n, \mathbb{R})$, dos número naturales $q, r \in \mathbb{N}$ y q números reales $\theta_i \in (0, \pi]$ tales que:

- $2q + r = n$
- $VUV^* = \mathcal{R}(\theta_1) \oplus \dots \oplus \mathcal{R}(\theta_q) \oplus Id_r$.

Definimos la matriz $M_{q,r} \in \mathcal{M}_n(\mathbb{R})$ por la suma diagonal:

$$M_{q,r} = M_{\theta_1} \oplus \dots \oplus M_{\theta_q} \oplus Id_r \in \mathcal{M}_n(\mathbb{C}),$$

donde M_{θ_i} son matrices cuadradas de dimensión 2 tales que:

$$M_{\theta_i} \mathcal{R}(\theta_i) M_{\theta_i}^t = \mathcal{J}(\theta_i).$$

Tenemos entonces que

$$M_{q,r} VUV^* M_{q,r}^{-1} = \mathcal{J}(\theta_1) \oplus \dots \oplus \mathcal{J}(\theta_q) \oplus Id_r.$$

Sea $P \in \text{GL}(n, \mathbb{C})$ una matriz regular con coeficientes complejos tal que $P = M_{q,r}V$. Finalmente, definimos la matriz

$$\mathcal{L} = D(\theta_1) \oplus \dots \oplus D(\theta_q) \oplus Id_r,$$

donde

$$D(\theta_j) = \begin{pmatrix} \theta_j & 0 \\ 0 & \theta_j \end{pmatrix},$$

para todo $j, 1 \leq j \leq q$. La matriz $L \in \mathcal{M}_n(\mathbb{R})$ dada por la siguiente igualdad $L = P^{-1}\mathcal{L}P$ cumple que

$$e^{iL} = U.$$

Es más, para todo $t \in [0, 1]$, la matriz e^{iLt} es una matriz regular tal que

$$e^{iLt} = V^t M_{q,r}^t e^{i\mathcal{L}t} M_{q,r} V,$$

y por lo tanto

$$e^{iLt} = V^t (R(\theta_1 t) \oplus \dots \oplus R(\theta_q t) \oplus Id_r) V,$$

es una matriz ortogonal con $\det(e^{iLt}) = 1$ tal y como se buscaba. \square

Corolario 6.0.7. *Usando la notación anterior. Dados $(f, x), (g, y)$ dos puntos en $V^{\mathbb{R}^+}$ se tiene que el siguiente camino*

$$\begin{aligned} \Gamma : [0, 1] &\longrightarrow V^{\mathbb{R}^+} \\ t &\mapsto (f \circ \gamma(t))^{-1}, \gamma(t)x, \end{aligned}$$

es un camino contenido en $V^{\mathbb{R}^+}$ que une (f, x) con (g, y) siempre que:

- $\gamma(1)x = y$.
- $\gamma(1) \in \text{SO}(n, \mathbb{R})$ y $g = f \circ \gamma(1)^{-1}$.

Es más, el condicionamiento μ_{norm} a lo largo del camino cumple

$$\mu_{norm}(\Gamma(t)) = \mu_{norm}(f, x) \quad \forall t \in [0, 1].$$

Capítulo 7

Trabajos Futuros

*However impenetrable it seems, if you don't try it, then you can never do it*¹.

Sir Andrew John Wiles [onl]

En este capítulo introduciremos algunas de las ideas que consideramos interesantes explorar en un futuro. Entre ellas comentamos los pasos que faltan por dar para terminar de desarrollar un algoritmo que siga el programa iniciado por M. Shub y S. Smale y continuado por C. Beltrán y L. M. Pardo. Finalmente daremos algunas alternativas que nos parecen interesantes y que trataremos de abordar en el futuro.

El mayor problema que se presenta es el excesivo número de componentes conexas que tiene el espacio de sistemas factibles como vimos en la Proposición 0.0.28. Para solventar este inconveniente tenemos diversas posibilidades:

- Desarrollar un test que nos indique cuantas raíces reales tiene un sistema de ecuaciones polinomiales homogéneas multivariadas. Como hemos visto en el Teorema 5.3.5 esto equivale a saber si dos sistemas están en la misma componente conexa.

Desgraciadamente no parece muy factible encontrar un algoritmo que resuelva este problema en tiempo polinomial pues es ya un problema muy duro en el caso unidimensional.

- Aun en el caso de tener dicho test debemos de encontrar un camino que una ambos sistemas sin abandonar la componente conexa. Este problema es relativamente más sencillo de resolver pues el condicionamiento μ_{norm} nos indica la proximidad a los sistemas singulares y por lo tanto a las fronteras entre componentes conexas. En caso de aproximarnos peligrosamente a los sistemas singulares basta con elegir de forma aleatoria una matriz ortogonal y *desviarse* en esa dirección.
- Otro problema a afrontar será conseguir que dicho camino sea *corto*, pues de nada serviría tenerlo si debemos dar muchos pasos para poder seguirlo. En este sentido en [BS09] se estudian las geodésicas en la métrica del condicionamiento. Encontrar una cota inferior al número de pasos de una sucesión de Newton que fuera exponencial proporcionaría una sentencia de muerte a este programa.

¹Por muy impenetrable que parezca, si no lo intentas, nunca lo conseguirás.

- Por último sería necesario crear un conjunto cuetor de sistema de ecuaciones polinomiales homogéneas multivariadas que contenga suficientes sistemas en cada una de las componentes conexas.

En el Capítulo 12 desarrollaremos una alternativa evolutiva para resolver este problema. Sin embargo no es la única posible solución. El principal resultado de [BP] se puede expresar como sigue:

Teorema 7.0.8 (Entropía). *La entropía de un método de deformación homotópica es máxima. Es decir, la probabilidad de que seleccionada al azar un sistema inicial un método de deformación homotópica produzca una raíz particular del sistema es \mathcal{D}^{-1} .*

A partir de este resultado queda claro que es posible encontrar varias raíces de un sistema de ecuaciones polinomiales homogéneas multivariadas siguiendo varios caminos simultáneamente, comenzando cada uno desde un sistema distinto, pues con alta probabilidad los caminos llevarán a soluciones distintas. Así, si por alguna razón se quiere conocer alguna raíz en especial, probando con distintos caminos, en algún momento se encontrará la raíz deseada.

En el caso de sistema de ecuaciones polinomiales homogéneas multivariadas con coeficientes reales ya hemos visto como el número de raíces es en media $\sqrt{\mathcal{D}}$ y por lo tanto el número de intentos se puede acotar superiormente. Desgraciadamente esta cota es del orden de \mathcal{D} y por lo tanto resulta un algoritmo intratable en la práctica.

Gran parte de estos problemas no se presentan en el caso que $m < n$ pues como ya hemos visto en el Teorema 0.0.27 independientemente de los valores de m y n se tiene que el número de componentes conexas es uno. Es más, como la codimensión de la variedad singular $\Sigma^{\mathbb{R}}$ es 2 se sigue que la homotopía lineal, con probabilidad 1, no corta la variedad singular.

Sin embargo, no hemos conseguido encontrar una cota al número de pasos necesarios para seguir dicha curva. Es decir no conocemos un resultado del estilo:

Conjetura 7.0.9 (Teorema de continuación homotópica). *Sea $m < n$ y $f, g \in \mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}}) \setminus \Sigma^{\mathbb{R}}$ dos sistema de ecuaciones polinomiales homogéneas multivariadas con coeficientes reales. Entonces existe un número natural k , polinomial en n, m, N y D , tal que:*

$$x_0 := \zeta \in V^{\mathbb{R}}(f_0), \quad x_{i+1} := N_{f_i}(x_i),$$

es una sucesión de Newton. Es decir, para todo i , $0 \leq i \leq k$,

$$x_i \text{ es un cero aproximado de algún } \zeta_i \in V^{\mathbb{R}}(f_i).$$

Sin este resultado no tiene sentido buscar intentar la construcción de un conjunto cuetor.

Parte II

Algoritmos Bioinspirados

Capítulo 8

Una *vieja* estructura de datos para un *nuevo* problema

*The programmer's primary weapon in the never-ending battle against slow system is to change the intramodular structure. Our first response should be to reorganize the modules' data structures*¹.

Fred Brooks [McC04]

Anteriormente hemos definido la programación genética como un algoritmo evolutivo en el que la estructura de datos a evolucionar son programas informáticos. Generalmente se suele usar este esquema algorítmico cuando se quiere resolver un proceso de *aprendizaje inductivo*. Típicamente estos programas se representan mediante expresiones en algún dialecto de LISP o mediante árboles dirigidos (véase [Koz92] para una extensa introducción).

Recientemente otras variantes han aparecido. Dichas variantes sustituyen las estructuras clásicas a evolucionar por otras como listas de instrucciones en un lenguaje imperativo. A este nuevo paradigma se le suele conocer como programación genética lineal. En esta ocasión el término *lineal* se refiere a que la estructura de datos usada consiste en una lista de asignaciones, operaciones sobre constantes o sobre variables. Mediante este mecanismo tan simple es posible obtener expresiones no lineales muy complejas. Los primeros textos al respecto se deben a Cramer [Cra85], otros desarrollos más modernos se deben a Banzhaf [Ban93] y Nordin [Nor94]. Una referencia introductoria completa al tema se puede encontrar en [BB07].

En este capítulo analizamos en detalle una estructura de datos para representar programas en el paradigma de la programación genética lineal: los Stright-Line Program (*slp*). Esta estructura de datos puede considerarse como representaciones lineales de programas, pero también como representaciones en forma de *grafos* (véase la Sección 8.1 para más detalles). Su uso como modelo teórico en el contexto de la Álgebra Computacional es ampliamente conocido, como se ha podido ver en los capítulos de introducción. Nuestra

¹La principal arma del programador en su batalla sin final contra los sistemas lentos es cambiar la estructura intramodular. Nuestra primera respuesta debería ser reorganizar las estructuras de datos de los módulos.

principal aportación en este capítulo es su reutilización en un contexto completamente diferente.

En la Sección 8.2.3 introduciremos los operadores genéticos desarrollados *ad-hoc* para esta estructura de datos con el fin de resolver instancias del problema de regresión simbólica. Además en la Sección 8.3 mostraremos una serie de resultados experimentales en el que demostramos que el desempeño de la programación genética en el problema de regresión simbólica es muy superior al usar *slp*'s como estructura de datos frente a usar árboles.

8.1 La estructura de datos *slp*

Sea $\mathcal{F} = \{f_1, \dots, f_n\}$ un conjunto de funciones donde para todo i , $1 \leq i \leq n$, f_i tiene aridad a_i . Sea $\mathcal{T} = \{t_1, \dots, t_m\}$ un conjunto de terminales que se puede descomponer en dos conjuntos $\mathcal{T} = V \cup C$ con

- $V = \{x_1, \dots, x_p\}$ un conjunto finito de variables.
- $C = \{c_1, \dots, c_q\}$ un conjunto finito de constantes.

Definición 8.1.1 (*slp*). Llamaremos *slp* sobre los conjuntos \mathcal{F} y \mathcal{T} a toda secuencia finita de instrucciones $\Gamma = \{I_1, \dots, I_l\}$ tales que:

$$I_k \equiv u_k := f_{j_k}(\alpha_1, \dots, \alpha_{a_{j_k}})$$

en donde:

- Para todo k , $f_{j_k} \in \mathcal{F}$.
- Si $k = 1$ entonces $\alpha_i \in \mathcal{T}$.
- Para $2 \leq k \leq l$, $\alpha_i \in \mathcal{T} \cup \{u_1, \dots, u_{k-1}\}$.

Decimos que el número de instrucciones l es la *longitud* de Γ y denotaremos por $SLP_l(\mathcal{F}, \mathcal{T})$ el conjunto de todos los posible *slp* que se pueden construir sobre el conjunto de funciones \mathcal{F} y el conjunto de terminales \mathcal{T} .

Nota 8.1.2. Si consideramos un *slp* Γ como el código de un programa, en cada instrucción I_i se introduce una nueva variable que no es un terminal. Así, el número de estas variables coincide con el número de instrucciones así como con la longitud del *slp*. Por lo tanto, en lo siguiente denotaremos un *slp* $\Gamma = \{I_1, \dots, I_l\}$ como $\Gamma = \{u_1, \dots, u_l\}$.

Cada uno de los u_i se puede considerar como un *sub-slp* sobre el conjunto de terminales \mathcal{T} construido como una secuencia de llamadas recursivas que evalúan una función de \mathcal{F} .

Ejemplo 8.1.3. Sea \mathcal{F} el conjunto de las operaciones aritméticas básicas $\mathcal{F} = \{+, -, *\}$ y sea $\mathcal{T} = \{1, x, y\}$ el conjunto de terminales. En esta situación todos los *slp* $\Gamma \in$

$SLP_l(\mathcal{F}, \mathcal{T})$ que se pueden definir son polinomios en dos variables con coeficientes enteros. Consideramos el siguiente *slp* de longitud 5:

$$\Gamma \equiv \begin{cases} u_1 := x + 1 \\ u_2 := u_1 * u_1 \\ u_3 := y + y \\ u_4 := u_2 * u_3 \\ u_5 := u_4 - u_3 \end{cases} \quad (8.1)$$

Entonces, la función que se computa al evaluar u_5 es el polinomio:

$$2y(x + 1)^2 - 2y.$$

El siguiente resultado calcula el cardinal del conjunto $SLP_l(\mathcal{F}, \mathcal{T})$, es decir nos da la cantidad de *slp*'s de una longitud dada l . Obviamente esta cantidad depende del cardinal de los conjuntos \mathcal{F} y \mathcal{T} de la aridad de las funciones \mathcal{F} .

Proposición 8.1.4. *Sea $\mathcal{F} = \{f_1, \dots, f_n\}$ el conjunto de funciones donde f_i tiene aridad a_i , $i = 1 \dots n$ y sea $\mathcal{T} = \{t_1, \dots, t_m\}$ el conjunto de terminales. Entonces, el número de elementos $\Gamma = \{u_1, \dots, u_l\} \in SLP_l(\mathcal{F}, \mathcal{T})$ de longitud l es:*

$$\prod_{j=1}^l \left(\sum_{i=1}^n (m + j - 1)^{a_i} \right).$$

Demostración. Para la primera variable no terminal u_1 existen $\sum_{i=1}^n m^{a_i}$ posibilidades. Para u_2 existen $\sum_{i=1}^n (m + 1)^{a_i}$ posibilidades pues u_1 puede aparecer como argumento de la función en u_2 . En general, hay $\sum_{i=1}^n (m + k - 1)^{a_i}$ posibilidades para las variables no terminales u_k , $1 \leq k \leq l$. Luego, el número de posibles *slp* sobre \mathcal{F} y \mathcal{T} de longitud l es:

$$\prod_{j=1}^l \left(\sum_{i=1}^n (m + j - 1)^{a_i} \right).$$

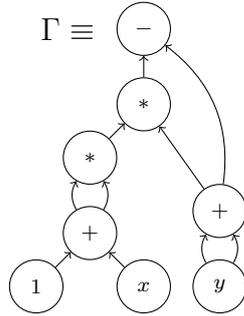
□

Nota 8.1.5. *Todo *slp* $\Gamma = \{u_1, \dots, u_l\}$ sobre los conjuntos \mathcal{F} y \mathcal{T} se puede representar como un grafo dirigido $G_\Gamma = (V, E)$. El conjunto de vértices es $V = \mathcal{T}' \cup \{u_1, \dots, u_l\}$ donde \mathcal{T}' contiene todos los terminales involucrados en la computación. El conjunto de aristas E se construye de la siguiente manera:*

*Para todo k , $1 \leq k \leq l$, dibujamos una arista (u_k, α_i) para cada $i \in \{1, \dots, a_{j_k}\}$. Nótese que \mathcal{T}' es el conjunto de hojas del grafo G_Γ y es un subconjunto del conjunto de terminales \mathcal{T} . La Figura 8.1 representa el grafo dirigido asociado al *slp* descrito en la Ecuación (8.1).*

La aplicación semántica asociada a un *slp* $\Gamma = \{u_1, \dots, u_l\}$ definido sobre el conjunto de funciones \mathcal{F} y el conjunto de terminales \mathcal{T} se construye definiendo en Γ el conjunto de variables de salida $O(\Gamma) = \{u_{i_1}, \dots, u_{i_t}\}$, entre cualquiera de las variables no terminales de

Figura 8.1: Grafo dirigido que representa un *slp*.



Γ . Teniendo en cuenta que $V = \{x_1, \dots, x_p\} \subset \mathcal{T}$ es el conjunto de variables, la aplicación semántica que representa Γ , que denotaremos como $\Phi_\Gamma : I^p \rightarrow O^t$, satisface

$$\Phi_\Gamma(a_1, \dots, a_p) = (b_1, \dots, b_t),$$

donde b_j representa los valores de las expresiones de las variables no terminales u_{i_j} cuando sustituimos en ellas las variables x_k por sus valores a_k , $1 \leq k \leq p$.

Dados dos *slp*'s Γ_1 y Γ_2 sobre los conjuntos \mathcal{F} y \mathcal{T} diremos que son equivalentes si calculan la misma aplicación semántica. Es decir:

$$\Phi_{\Gamma_1} \equiv \Phi_{\Gamma_2}.$$

Sea $\Gamma = \{u_1, \dots, u_l\}$ un *slp* sobre los conjuntos \mathcal{F} y \mathcal{T} con conjunto de salida $O(\Gamma) = \{u_{i_1}, \dots, u_{i_t}\}$. Asumiendo que $1 \leq i_1 < \dots < i_t \leq l$ es fácil ver que el *slp* $\Gamma' = \{u_1, \dots, u_{i_t}\}$ es equivalente a Γ . Nótese que para el cálculo de la aplicación semántica Φ_Γ son necesarios como mucho las variables no terminales u_1, \dots, u_{i_t} . Luego

$$\Phi_\Gamma \equiv \Phi_{\Gamma'}.$$

Es por esto que a partir de ahora supondremos, sin pérdida de generalidad, que el conjunto de salida de un *slp* siempre contendrá la última variable no terminal. Es decir,

$$u_l \in O(\Gamma).$$

Nota 8.1.6. Nótese que, mediante la relación de equivalencia de *slp* descrita anteriormente, podemos identificar el espacio de las aplicaciones semánticas con el conjunto de *slp*'s.

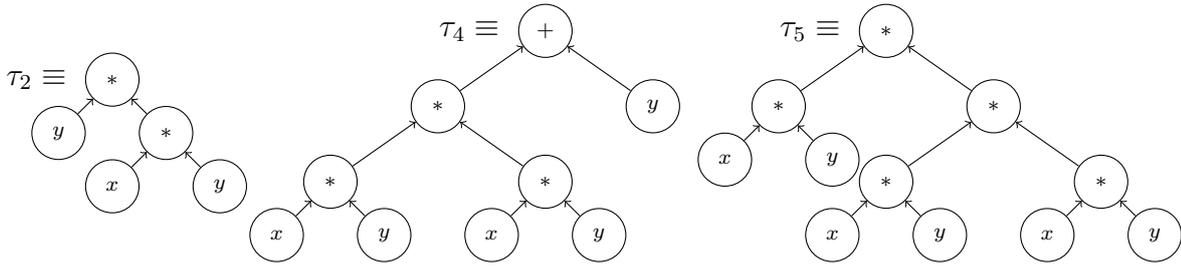
8.1.1 Slp's frente a árboles

En esta sección presentaremos las relaciones que naturalmente aparecen al comparar los *slp* con los árboles.

Definición 8.1.7. Sea $\Gamma = \{u_1, \dots, u_l\}$ un *slp* sobre los conjuntos \mathcal{F} y \mathcal{T} . Para cada $k \in \{1, \dots, l\}$ definimos el árbol τ_k asociado a la variable no terminal u_k como:

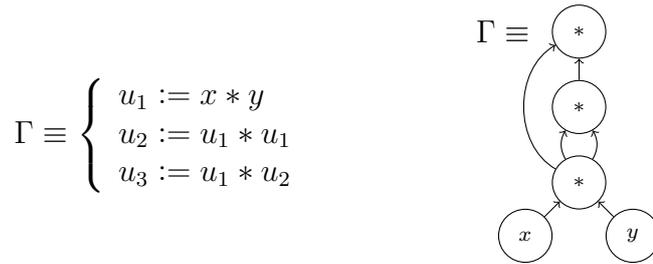
- Etiquetamos la raíz del árbol τ_k como f_{j_k} .

Figura 8.2: Bosque asociado al *slp* Γ del Ejemplo 8.1.10.



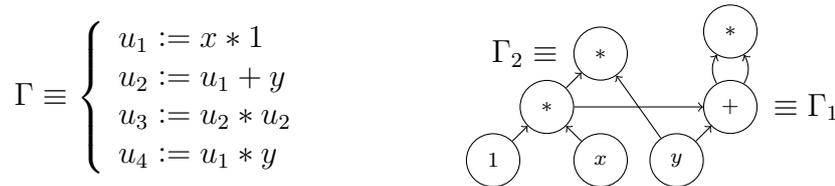
La construcción del *slp* $\Gamma(\tau)$ se realiza identificando las instrucciones u_i , $1 \leq i \leq l$, recursivamente a partir de las hojas del árbol τ hasta la raíz. Siguiendo este proceso es claro que la longitud l del *slp* $\Gamma(\tau)$ siempre será mayor o igual que la altura del árbol τ . Por otro lado, claramente se observa que el bosque asociado al *slp* $\Gamma(\tau)$ es τ . Nótese que no se asegura nada acerca de la relación entre la cantidad de nodos del árbol τ y la longitud del *slp*.

El siguiente es un ejemplo que surge al construir el *slp* asociado al árbol τ_5 de la Figura 8.2.

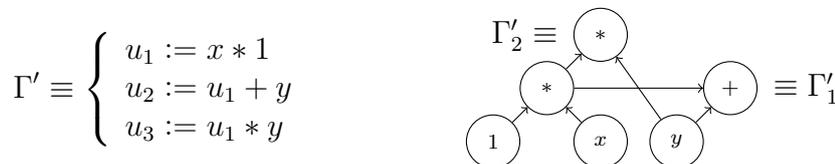


8.1.2 Código efectivo y no efectivo de un *slp*

Consideremos el siguiente *slp* Γ sobre los conjuntos $\mathcal{F} = \{+, *, -\}$ y $\mathcal{T} = \{1, x, y\}$,



Sea $O(\Gamma) = \{u_2, u_4\}$ el conjunto de salida de Γ . Nótese que si queremos calcular el valor de la aplicación semántica que define Γ para un valor $(a_1, a_2) \in \mathbb{R}^2$ no es necesario calcular el valor de la variable no terminal u_3 . Diremos en este caso que la variable no terminal u_3 es *código no efectivo* en Γ y podría ser eliminado si sólo estuviéramos interesados en la aplicación semántica que define Γ . Tras eliminar u_3 y renombrar el resto de variables no terminales de Γ , obtenemos el siguiente *slp* Γ' :

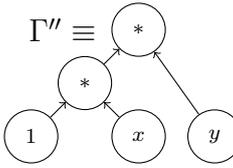


Γ' es equivalente a Γ pues ambos calculan la misma aplicación semántica:

$$\begin{aligned} \Phi : \mathbb{R}^2 &\longrightarrow \mathbb{R}^2 \\ (x, y) &\longmapsto (x + y, xy) \end{aligned}$$

cuando tomamos como conjunto de salida de Γ' las variables no terminales $O(\Gamma') = \{u_2, u_3\}$.

Ahora bien, si el conjunto de salida del *slp* Γ fuera el conjunto $O(\Gamma) = \{u_4\}$, las variables no terminales u_2 y u_3 pasarían a ser código no efectivo y ahora el *slp* Γ'' sería equivalente a Γ :

$$\Gamma'' \equiv \begin{cases} u_1 := x * 1 \\ u_2 := u_1 * y \end{cases}$$


donde $O(\Gamma'') = \{u_2\}$.

Definición 8.1.11 (Código efectivo). *En general, definimos el código efectivo de un *slp* $\Gamma = \{u_1, \dots, u_l\}$ con conjunto de salida $O(\Gamma) = \{u_{i_1}, \dots, u_{i_t} = u_l\}$ como el conjunto de variables no terminales que son necesarias para evaluar la aplicación semántica Φ_Γ . Denotaremos a este conjunto como:*

$$S = \{u_j \in \Gamma : \exists u_{i_k} \in O(\Gamma) u_j \prec u_{i_k}\} = \{u_{j_1}, \dots, u_{j_m}\},$$

donde \prec es la relación de orden parcial de la Definición 8.1.8.

Asumiendo que $j_1 < \dots < j_m$, para obtener S construimos una cadena no decreciente de conjuntos

$$S_0 \subseteq S_1 \subseteq \dots \subseteq S_p = S$$

donde $S_0 = O(\Gamma)$ y

$$S_k = S_{k-1} \cup \{u_i \in \Gamma : \exists u_j \in S_{k-1} u_i \prec u_j\},$$

donde \prec es, otra vez, la relación de orden parcial de la Definición 8.1.8.

Es fácil comprobar que la cadena de conjuntos se vuelve estacionaria en un número finito de pasos. En el peor de los casos S_p es Γ . Además, claramente $u_{j_m} = u_l$ y siempre es posible construir un nuevo *slp* $\Gamma' = \{u'_1, \dots, u'_m\}$ considerando las variables no terminales en S y usando la aplicación R sobre S tal que:

$$R(u_{j_k}) = u'_k.$$

Nótese que Γ' es equivalente a Γ si tomamos como conjunto de salida para Γ'

$$O(\Gamma') = \{R(u_{i_1}), \dots, R(u_{i_t})\}.$$

Además Γ' satisface

$$\forall i \in \{1, \dots, m\} \exists u'_j \in O(\Gamma') : u'_i \prec u'_j.$$

Finalmente, si $\Gamma = \Gamma'$ diremos que Γ es un *slp* efectivo.

8.1.3 Cálculo de la aplicación semántica

Sea \mathcal{F} un conjunto de funciones y \mathcal{T} un conjunto de terminales con conjunto de variables $V = \{x_1, \dots, x_n\}$. La estrategia para calcular la aplicación semántica de un *slp* $\Gamma = \{u_1, \dots, u_t\}$ sobre \mathcal{F} y \mathcal{T} consistente en evaluar las variables no terminales siguiendo el orden de declaración en Γ , no es óptima cuando Γ no es un *slp* efectivo. En la práctica, es mejor obtener el *slp* efectivo equivalente a Γ y posteriormente evaluar este. El siguiente algoritmo describe este método:

Algoritmo 8.1.12 (Cálculo de la aplicación semántica).

Input: $\Gamma = \{u_1, \dots, u_t\}$ un *slp* con conjunto de salida $O(\Gamma) = \{u_{i_1}, \dots, u_{i_t}\}$.
 (a_1, \dots, a_n) donde a_i es el valor de la variable x_i .

Output: $\Phi_\Gamma(a_1, \dots, a_n) = (b_1, \dots, b_t)$.

1. Computar el conjunto $S = \{u_{j_1}, \dots, u_{j_m}\}$, $j_1 < \dots < j_m$, descrito anteriormente mediante los conjuntos parciales S_k . Nótese que $\Gamma' = \{R(u_{j_1}), \dots, R(u_{j_m})\}$ es un *slp* efectivo equivalente a Γ .
2. Para $k = 1$ hasta m evaluar $R(u_{j_k}) = u'_k$ reemplazando cada ocurrencia de x_i por a_i y cada ocurrencia de u'_j , $j < k$ por su valor que ha sido previamente calculado.
3. Devolver el vector de valores $(R(u_{i_1}), \dots, R(u_{i_t}))$.

8.2 Programación genética con *slp* en problemas de regresión

El problema de regresión simbólica consiste en, dada una muestra de datos, encontrar la aplicación simbólica en un espacio de búsqueda determinado que mejor aproxima la muestra de datos. Más formalmente, consideramos el conjunto de entrada $X = \mathbb{R}^n$ y salida $Y = \mathbb{R}^l$.

Sea $z = \{(x_i, y_i), 1 \leq i \leq m\}$ un conjunto de m muestras siguiendo una distribución de probabilidad desconocida ρ en el espacio producto $Z = X \times Y$ independientes e igualmente distribuidas (i.i.d.). El objetivo es encontrar un aplicación $f : X \rightarrow Y$ que prediga el valor $y \in Y$ dado un valor $x \in X$. El objetivo es disminuir la probabilidad de error por lo que el criterio para escoger dicha aplicación f se basa en minimizar el *MSE*:

$$\varepsilon_z(f) = \frac{1}{m} \sum_{i=1}^m \|f(x_i) - y_i\|^2.$$

Los problemas de regresión simbólica se puede resolver usando programación genética de diversas formas. Usualmente se trata de evolucionar estructuras similares a árboles que de alguna forma codifican aplicaciones. Estas estructuras sufren luego un proceso evolutivo siguiendo los principios *darwinianos* de supervivencia y reproducción del más «apto» o mejor cualificado. Sin embargo en esta memoria se usará como estructura para codificar estas aplicaciones los *slp* descritos en la Sección 8.1. La principal ventaja que presentan

estas estructuras con respecto a los árboles es su capacidad para reusar los resultados de una subexpresión con el fin de evitar cálculos innecesarios. Esto trae como consecuencia inmediata que el tamaño necesario para codificar expresiones complejas se reduce. Otra de las ventajas es el control sobre el tamaño de las estructuras. En efecto, gracias al código no efectivo, se puede forzar que todas las estructuras tengan el mismo tamaño. De todas maneras, sin unos operadores genéticos adecuados que exploten correctamente estas características poco beneficio se puede extraer de esta estructura.

En un lenguaje de muy alto nivel, el proceso evolutivo que propone la programación genética y que se ha implementado es el descrito en la introducción (véase el Pseudocódigo 3).

A lo largo de esta memoria el conjunto de salida de los *slp* $\Gamma = \{u_1, \dots, u_l\}$ siempre estará compuesto por un único elemento, es decir, $O(\Gamma) = \{u_l\}$. Por lo tanto, los *slp*'s siempre computarán funciones reales multivariadas.

8.2.1 La población inicial

Sea $\mathcal{F} = \{f_1, \dots, f_n\}$ un conjunto de funciones tales que f_i tiene aridad a_i , para todo i , $1 \leq i \leq n$ y sea $\mathcal{T} = \{t_1, \dots, t_m\}$ un conjunto de terminales tal como hemos introducido en la Sección 8.1. A continuación describimos el proceso para generar cada *slp* de la población inicial.

Para la primera variable no terminal u_1 seleccionamos una función $f_{j_1} \in \mathcal{F}$ aleatoriamente. Para cada argumento $i \in \{1, \dots, a_{j_1}\}$ de esta función f_{j_1} escogemos aleatoriamente un elemento α_i del conjunto de terminales \mathcal{T} .

En general, para la variable no terminal u_k , $k > 1$, empezamos igualmente escogiendo una función $f_{j_k} \in \mathcal{F}$. La diferencia se encuentra ahora, para cada argumento i , $1 \leq i \leq a_{j_k}$ escogemos, aleatoriamente, un elemento en el conjunto $\mathcal{T} \cup \{u_1, \dots, u_{k-1}\}$.

En la práctica, hay que introducir una cota superior L para la longitud de los *slp*'s involucrados en el proceso de programación genética. Por lo tanto, el primer paso para la generación de cada *slp* debería consistir en la selección de forma aleatoria de la longitud $l \in \{1, \dots, L\}$.

Nótese que los *slp*'s generados de esta forma pueden ser no efectivos. De todas formas, nuestro objetivo es permitir este comportamiento durante el proceso evolutivo pues experimentalmente se observa una gran mejoría. Por otro lado, fijaremos la longitud de cada individuo de forma homogénea. Para ello, la longitud de los individuos pasará a ser un parámetro más del algoritmo. Nótese que este tipo de homogeneidad no es posible con poblaciones de árboles tradicionales.

8.2.2 El operador de calificación

En programación genética es necesario calificar de alguna manera cada individuo y a partir de este parámetro producir una simulación del proceso natural de evolución. La forma común de proceder es asignar un valor a partir de un operador de calificación, esto es, definiendo una función en el espacio de búsqueda. En [Koz92] se presentan algunas

alternativas para estas funciones que se han usado en el contexto de la programación genética. El primer operador de calificación que introduciremos es el error cuadrático medio (*MSE*), que consiste en evaluar el valor de la aplicación semántica de un *slp* a lo largo de todos los elementos de la muestra y hallar cuanto divergen respecto del valor esperado. Esto es, dados $z = (x_i, y_i) \in \mathbb{R}^n \times \mathbb{R}^l$, $1 \leq i \leq m$, para cualquier *slp* Γ sobre los conjuntos \mathcal{F} y \mathcal{T} definiremos el operador de calificación actuando sobre Γ como:

$$\mathcal{Fit}_z(\Gamma) = \varepsilon_z(\Phi_\Gamma) = \frac{1}{m} \sum_{i=1}^m \|\Phi_\Gamma(x_i) - y_i\|^2 \quad (8.2)$$

Esto es, la calificación es el error empírico de la aplicación semántica de Γ respecto de la muestra z .

8.2.3 Operadores genéticos

Operador de cruce

Debido a que codificamos los *slp's* como una secuencia finita de operaciones y teniendo en cuenta que todos los individuos tienen la misma longitud, los métodos clásicos de cruce tales como: el cruce en un punto, el cruce en dos puntos y el cruce uniforme pueden ser adaptados a esta estructura de datos fácilmente. A continuación se presentan los detalles:

Definición 8.2.1. *Sea $\Gamma_1 = \{u_1, \dots, u_L\}$ y $\Gamma_2 = \{u'_1, \dots, u'_L\}$ dos *slp's* de igual longitud sobre los conjuntos \mathcal{F} y \mathcal{T} . Entonces los operadores de cruce anteriormente citados se comportan sobre estas estructuras de la siguiente forma:*

En un punto: *Dado un punto de ruptura $i \in \{1, \dots, L\}$ el cruce en un punto operando sobre los *slp's* Γ_1 y Γ_2 produce*

$$\Gamma'_1 = \{u_1, \dots, u_i, u'_{i+1}, \dots, u'_L\}, \quad \Gamma'_2 = \{u'_1, \dots, u'_i, u_{i+1}, \dots, u_L\}.$$

Un ejemplo se puede encontrar en la Figura 8.4.

En dos puntos: *Dados dos punto de ruptura $i, j \in \{1, \dots, L\}$ con $i < j$ el cruce en dos puntos operando sobre los *slp's* Γ_1 y Γ_2 produce*

$$\begin{aligned} \Gamma'_1 &= \{u_1, \dots, u_i, u'_{i+1}, \dots, u'_j, u_{j+1}, \dots, u_L\}, \\ \Gamma'_2 &= \{u'_1, \dots, u'_i, u_{i+1}, \dots, u_j, u'_{j+1}, \dots, u'_L\}. \end{aligned}$$

Un ejemplo se puede encontrar en la Figura 8.5.

Uniforme: *Dado un vector de booleanos $\alpha = (\alpha_1, \dots, \alpha_L) \in \{0, 1\}^L$ el cruce uniforme operando sobre los *slp's* Γ_1 y Γ_2 produce*

$$\Gamma'_1 = \{v_1, \dots, v_L\}, \quad \Gamma'_2 = \{v'_1, \dots, v'_L\}$$

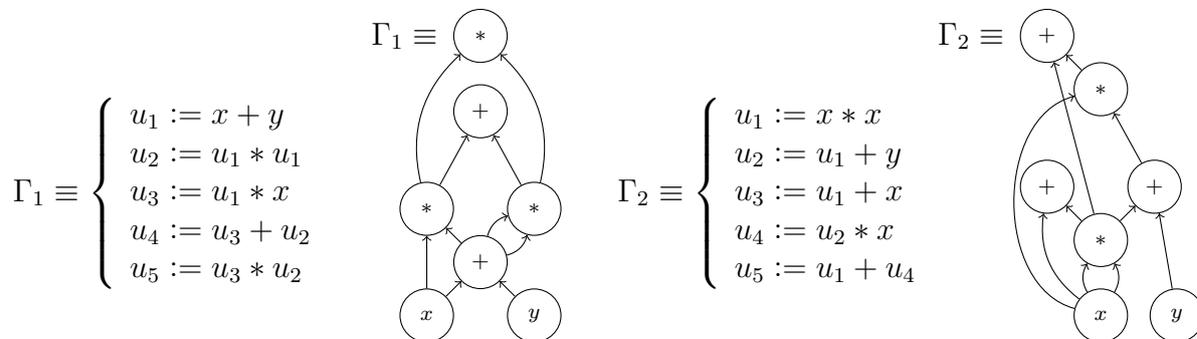
donde para todo $i \in \{1, \dots, L\}$ se cumple

$$\begin{aligned} \text{Si } \alpha_i = 0 \text{ entonces } v_i &= u_i \text{ y } v'_i = u'_i. \\ \text{Si } \alpha_i = 1 \text{ entonces } v_i &= u'_i \text{ y } v'_i = u_i. \end{aligned}$$

Un ejemplo se puede encontrar en la Figura 8.6.

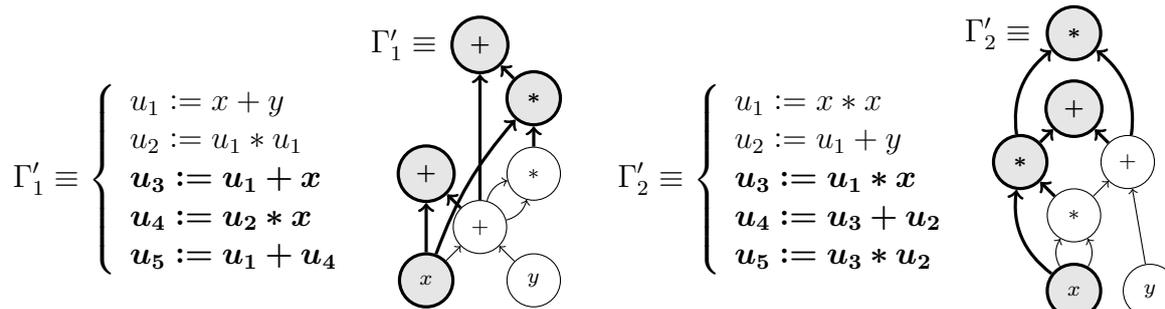
Ejemplo 8.2.2. Sea $\mathcal{F} = \{*, +\}$, $L = 5$ y $\mathcal{T} = \{x, y\}$. Consideramos ahora los siguientes slp's Γ_1 y Γ_2 :

Figura 8.3: Slp's padres.



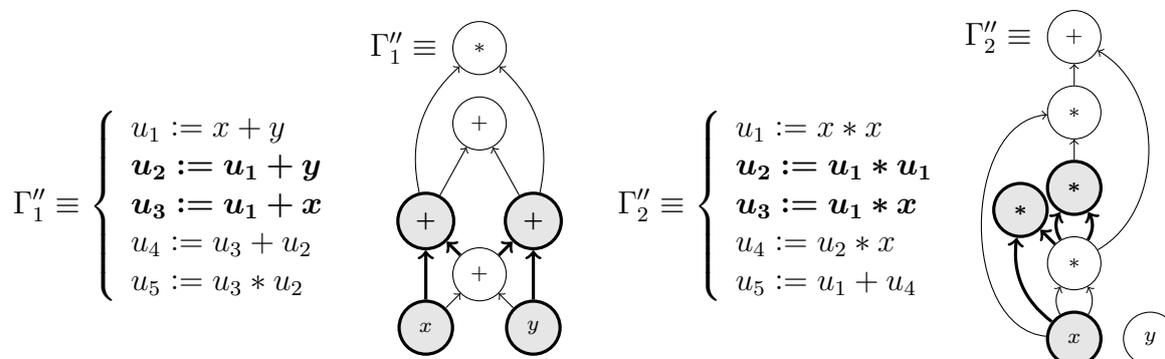
El cruce en un punto entre Γ_1 y Γ_2 con punto de ruptura $i = 2$ produce los slp's:

Figura 8.4: Slp's hijos mediante el cruce en un punto.



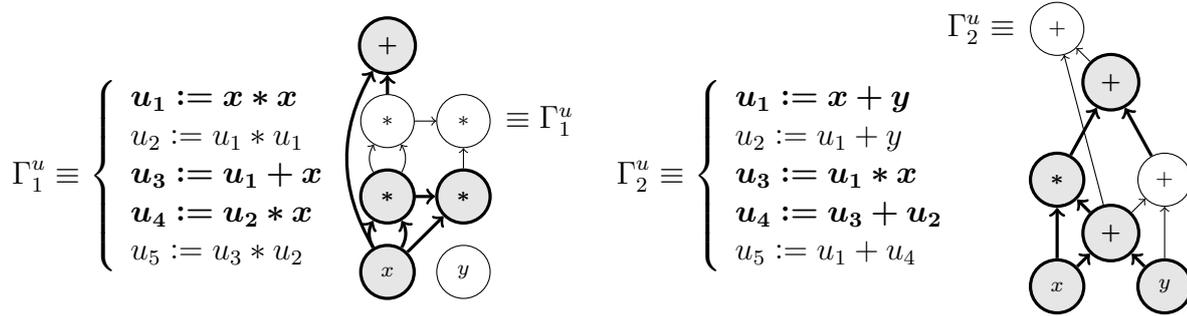
El cruce en dos puntos entre Γ_1 y Γ_2 con puntos de ruptura $i = 1$ y $j = 3$ produce los slp's:

Figura 8.5: Slp's hijos mediante el cruce en dos puntos.



Finalmente el cruce uniforme entre Γ_1 y Γ_2 con vector binario aleatorio $\alpha = (10110)$ produce los slp's:

Figura 8.6: *Slp's* hijos mediante el cruce uniforme.



La experimentación con dichos operadores de cruce no produce buenos resultados como veremos en la Sección 8.3. Esto es debido a que estos cruces destruyen las subexpresiones que codifican los *slp's* padres (esto se puede apreciar claramente en la representación gráfica de los cruces anteriores). Es por ello que se introduce un nuevo operador de cruce que trata de mantener e intercambiar, en lo posible, las subexpresiones de los *slp's* padres.

Una subexpresión está constituida por una variable no terminal u_i y todas las variables no terminales u_j relacionadas con u_i . En otras palabras, una subexpresión es el código efectivo asociado con la variable no terminal u_i . Formalmente, este operador de cruce opera de la siguiente forma.

Definición 8.2.3 (Cruce *slp*). Sea $\Gamma_1 = \{u_1, \dots, u_L\}$ y $\Gamma_2 = \{u'_1, \dots, u'_L\}$ dos *slp's* sobre los conjuntos \mathcal{F} y \mathcal{T} . Sea $k, 1 \leq k \leq L$, un número escogido al azar. Tal y como realizamos en la Definición 8.1.11 consideramos el conjunto

$$S_{u_k} = \{u_j \in \Gamma_1 : u_j \prec u_k\} = \{u_{j_1}, \dots, u_{j_m}\}$$

con $j_1 < \dots < j_m$. Tal y como hemos mencionado anteriormente, el conjunto S_{u_k} es el código efectivo de Γ_1 necesario para evaluar la variable no terminal u_k .

A continuación seleccionamos un número t tal que $m \leq t \leq L$. Este número representa la posición en Γ_2 en donde introduciremos el conjunto de variables no terminales S_{u_k} reemplazando las variables no terminales $\{u'_{t-m+1}, \dots, u'_t\}$ de Γ_2 . Si fuera necesario, se renombraría los índices mediante la siguiente aplicación

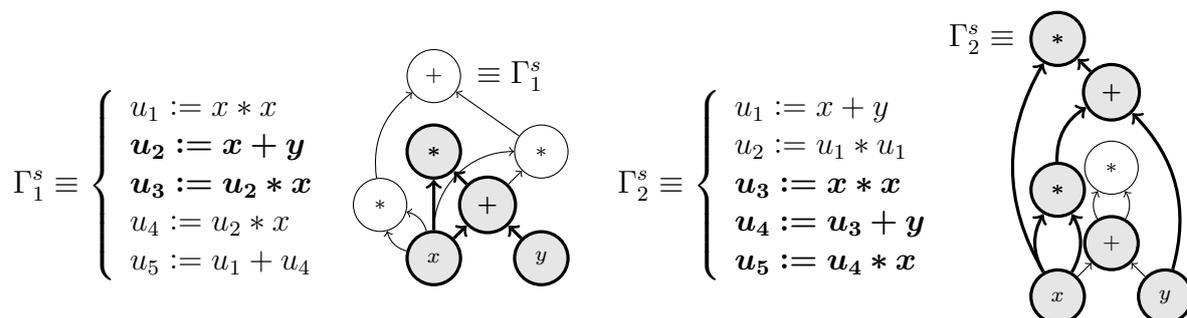
$$R : \begin{array}{l} S_{u_k} \longrightarrow \{u'_{t-m+1}, \dots, u'_t\} \\ u_{j_i} \longmapsto u'_{t-m+i} \end{array}$$

Mediante este proceso se obtiene el primer hijo producido a partir de los *slp's* Γ_1 y Γ_2 . Para producir el segundo hijo se aplica el mismo procedimiento simétricamente.

Ejemplo 8.2.4. Consideremos los *slp's* Γ_1 y Γ_2 procedentes del Ejemplo 8.2.2. Si $k = 3$ entonces $S_{u_3} = \{u_1, u_3\}$ y por lo tanto t debe escogerse en el conjunto $\{2, \dots, 5\}$. Supongamos que $t = 3$, entonces el primer *slp* hijo Γ_1^s producido por el cruce *slp* se puede observar en la Figura 8.7.

Para el segundo hijo, supongamos que hemos seleccionado la posición $k' = 4$ en Γ_2 . Luego $S_{u_4} = \{u_1, u_2, u_4\}$ y suponiendo que $t = 5$, el segundo *slp* hijo Γ_2^s se puede observar en la misma Figura 8.7.

Figura 8.7: *Slp's hijos mediante el cruce slp.*



Nota 8.2.5. *Obsérvese en la Figura 8.7 como, en este caso, el cruce slp mantiene siempre las subexpresiones (las flechas gruesas nunca parten de un nodo blanco). En contraposición, los cruces en un punto, en dos puntos y uniforme no siempre las conservan, como se puede observar en las Figuras 8.4, 8.5 y 8.6.*

Operadores de mutación, selección y reproducción

El operador de mutación es asexual y por lo tanto actúa sobre un único padre. Esta operación introduce un pequeño cambio aleatorio en el individuo. La mutación es beneficiosa pues ayuda a mantener la diversidad en la población que de otra manera tendería a converger prematuramente hacia un *óptimo local*. En la la Sección 8.3 veremos lo grave que puede resultar este problema.

El primer paso para aplicar la mutación slp a un *slp* Γ consiste en seleccionar una variable no terminal $u_i \in \Gamma$ de forma aleatoria. Posteriormente seleccionaremos, también de forma aleatoria, uno de los argumentos de la función $f \in F$ de la variable no terminal u_i . Finalmente, sustituiremos dicho argumento por uno en el conjunto $\mathcal{T} \cup \{u_1, \dots, u_{i-1}\}$ de forma aleatoria. Formalmente, la operación de mutación slp se describe:

Definición 8.2.6 (Mutación slp). *Sea $\Gamma = \{u_1, \dots, u_L\}$ un *slp* sobre los conjuntos \mathcal{F} y \mathcal{T} y sea $u_i = f(\alpha_1, \dots, \alpha_n)$ la variable no terminal seleccionada para ser mutada donde $f \in \mathcal{F}$ y $\alpha_k \in \mathcal{T} \cup \{u_1, \dots, u_{i-1}\}$. La mutación del *slp* Γ en el punto i produce:*

$$\Gamma' = \{u_1, \dots, u_{i-1}, u'_i, u_{i+1}, \dots, u_L\},$$

donde $u'_i = f(\alpha_1, \dots, \alpha_{j-1}, \alpha'_j, \alpha_{j+1}, \dots, \alpha_n)$, $\alpha_j \in \mathcal{T} \cup \{u_1, \dots, u_{i-1}\} \setminus \{\alpha'_j\}$ y tanto j como α'_j han sido seleccionados al azar.

La operación de reproducción consiste simplemente en la copia, sin modificación alguna, de un individuo a la siguiente población.

A lo largo de este manuscrito usaremos reemplazamiento entre poblaciones pero en el proceso de construcción de la nueva población introducimos un nuevo operador (el operador de reemplazo) que se encarga de seleccionar que individuos, de entre padres e hijos, pasan a la siguiente población. Este operador actúa de la siguiente manera:

Definición 8.2.7 (Operador de reemplazo). *Sean Γ_1 y Γ_2 dos *slp's* con calificaciones g_1 y g_2 . Sean Γ'_1 y Γ'_2 los *slp's* que resultan de haber realizado sobre Γ_1 y Γ_2 una operación*

de cruce. Supongamos que Γ'_1 y Γ'_2 tienen calificaciones g'_1 y g'_2 . Entonces el operador de reemplazo usará los dos mejores individuos, con calificaciones distintas, para introducirlos en la nueva población.

Nota 8.2.8. El operador de reemplazo funciona de la misma forma en el caso de una operación de mutación, sólo que selecciona un individuo de entre los dos posibles (el *slp* original y el mutado).

Mediante esta estrategia se intenta mantener la diversidad genética en la población que, como ya se ha mencionado, puede evitar la convergencia prematura hacia un óptimo local.

Siguiendo con este espíritu de evitar la convergencia prematura a un óptimo local, hay que tratar de evitar el uso de operadores de selección de individuos mediante proceso basados en la calificación como, por ejemplo, la selección ruleta. En vez de ello proponemos el uso del siguiente método, que es muy similar al método clásico q -torneo (véase [Koz92] para una descripción pormenorizada de éste y del anterior método de selección).

Definición 8.2.9 (Selección simple). Sea $(\Gamma_1, \dots, \Gamma_N)$ una población de *slp*'s. El proceso de selección simple escoge el elemento Γ_{v_i} o los elementos Γ_{v_i} y Γ_{v_i+1} dependiendo de si la operación genética a realizar es una mutación o un cruce y donde v_i representa la siguiente sucesión:

$$v_0 = k, \text{ donde } k \text{ es escogido al azar al comienzo de la generación.}$$

$$v_i = \begin{cases} v_{i-1} + 1 & \text{si la operación a realizar es una mutación.} \\ v_{i-1} + 2 & \text{si la operación a realizar es un cruce.} \end{cases}$$

Nótese que tomamos los elementos v_i en el grupo cíclico $\mathbb{Z}/N\mathbb{Z}$ donde N representa el número de individuos en la población.

En la Sección 8.3 realizaremos una comparativa entre el rendimiento de la programación genética basado en *slp*'s frente a la programación genética basada en árboles (otros resultados similares se pueden encontrar en [AMPB09]).

8.3 Resultados experimentales

En las siguientes páginas realizaremos un estudio empírico en el que mostraremos que, para el problema de regresión simbólica, la estructura de datos *slp* usando los operadores genéticos especialmente adaptados expuestos en la Sección 8.2.3 produce resultados estadísticamente significativos a su favor cuando se lo compara con los árboles. Para llevar a cabo dicho estudio seguiremos el protocolo propuesto en el Anexo B. Pasamos ahora a detallar los parámetros particulares de este experimento.

En primer lugar queremos comprobar que operador de cruce tiene mejor rendimiento al usar como estructura de datos los *slp*'s. Para ello realizaremos la experimentación comparando los cruces: en un punto, en dos puntos, uniforme y, por último, el cruce *slp*. Posteriormente usaremos dicho cruce para comprar que estructura de datos produce mejores resultados.

8.3.1 Operador de cruce para la estructura de datos *slp*

Los parámetros específicos de este experimento son:

Cuadro 8.1: Parámetros específicos: *cruces para slp*.

Tipo de estrategia:	Sólo programación genética
Estructura de datos:	<i>slp</i>
Operador de calificación:	<i>MSE</i>
Operador de cruce:	{en un punto, en dos puntos, uniforme, <i>slp</i> }
Operador de mutación:	<i>slp</i>

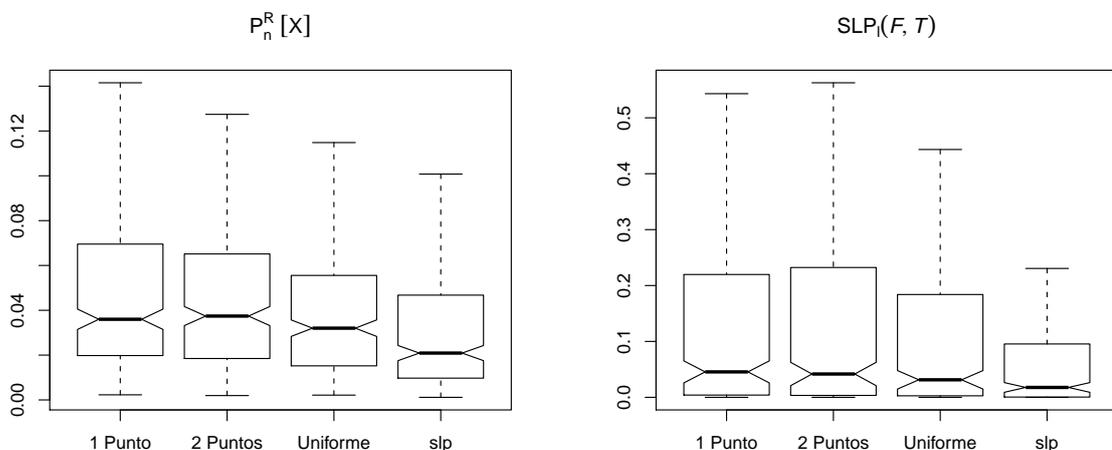
En la Tabla 8.2 podemos observar que tanto la tasa de error (t_e) como la tasa de éxito (t_s) son muy similares entre todos los cruces con ambas muestras. Aunque es bastante evidente que los resultados son peores en el caso de muestras generadas sobre *slp*'s aleatorios. La explicación para este fenómeno es que las muestras que se obtienen mediante este método son *mucho* más complejas y menos regulares y por lo tanto susceptibles de cometer errores al seleccionar el modelo.

Cuadro 8.2: Tasas de error y de solución: *cruces para slp*.

$P_n^R[X]$	t_e	t_s	$SLP_l(\mathcal{F}, \mathcal{T})$	t_e	t_s
1 Punto	7,33 %	100 %	1 Punto	13,67 %	90,33 %
2 Puntos	8,67 %	99,67 %	2 Puntos	16,33 %	91,33 %
Uniforme	8 %	100 %	Uniforme	17,67 %	93,67 %
<i>slp</i>	7 %	100 %	<i>slp</i>	15,33 %	96,33 %

Sin embargo cuando observamos los diagramas de cajas sobre ambas muestras (Figura 8.8) podemos observar una diferencia bastante significativa entre las medianas del cruce *slp* y el resto.

Figura 8.8: Diagramas de Cajas y Bigotes: *cruces para slp*.



8. Una *vieja* estructura de datos para un *nuevo* problema

Este hecho también se observa en las Tablas 8.3 y 8.4 pues todos los estadísticos son inferiores en el caso del cruce slp. Un hecho significativo lo observamos en la columna referida al mínimo en las muestras generadas mediante *slp*'s aleatorios. En alguna ocasión el procedimiento ha conseguido dar con el modelo *exacto*.

Cuadro 8.3: Resumen estadístico I: *cruces para slp*.

$P_n^R[X]$	μ	σ	$SLP_l(\mathcal{F}, \mathcal{T})$	μ	σ
1 Punto	$4,23 \cdot 10^{-2}$	$3,12 \cdot 10^{-2}$	1 Punto	$9,64 \cdot 10^{-2}$	0,15
2 Puntos	$4,05 \cdot 10^{-2}$	$2,94 \cdot 10^{-2}$	2 Puntos	$7 \cdot 10^{-2}$	$9,65 \cdot 10^{-2}$
Uniforme	$3,47 \cdot 10^{-2}$	$2,55 \cdot 10^{-2}$	Uniforme	$5,39 \cdot 10^{-2}$	$8,68 \cdot 10^{-2}$
slp	$2,62 \cdot 10^{-2}$	$2,27 \cdot 10^{-2}$	slp	$3,26 \cdot 10^{-2}$	$5,08 \cdot 10^{-2}$

Cuadro 8.4: Resumen estadístico II: *cruces para slp*.

$P_n^R[X]$	mín	q_1	q_2	q_3	máx
1 Punto	$2,28 \cdot 10^{-3}$	$1,83 \cdot 10^{-2}$	$3,32 \cdot 10^{-2}$	$5,87 \cdot 10^{-2}$	0,14
2 Puntos	$1,94 \cdot 10^{-3}$	$1,76 \cdot 10^{-2}$	$3,38 \cdot 10^{-2}$	$5,46 \cdot 10^{-2}$	0,13
Uniforme	$2,14 \cdot 10^{-3}$	$1,4 \cdot 10^{-2}$	$2,88 \cdot 10^{-2}$	$5,06 \cdot 10^{-2}$	0,11
slp	$1,15 \cdot 10^{-3}$	$9,34 \cdot 10^{-3}$	$1,91 \cdot 10^{-2}$	$3,76 \cdot 10^{-2}$	0,1

$SLP_l(\mathcal{F}, \mathcal{T})$	mín	q_1	q_2	q_3	máx
1 Punto	0	$3,29 \cdot 10^{-3}$	$2,97 \cdot 10^{-2}$	0,1	0,66
2 Puntos	0	$1,85 \cdot 10^{-3}$	$2,83 \cdot 10^{-2}$	0,1	0,46
Uniforme	0	$9,3 \cdot 10^{-4}$	$1,82 \cdot 10^{-2}$	$6,76 \cdot 10^{-2}$	0,44
slp	0	$4,19 \cdot 10^{-5}$	$9,61 \cdot 10^{-3}$	$4,09 \cdot 10^{-2}$	0,23

Pero es más, los contrastes de hipótesis no dejan lugar a dudas. Para cualquier valor razonable de la significancia α el cruce slp se muestra significativamente superior al resto.

Cuadro 8.5: Contraste de Hipótesis: *cruces para slp*.

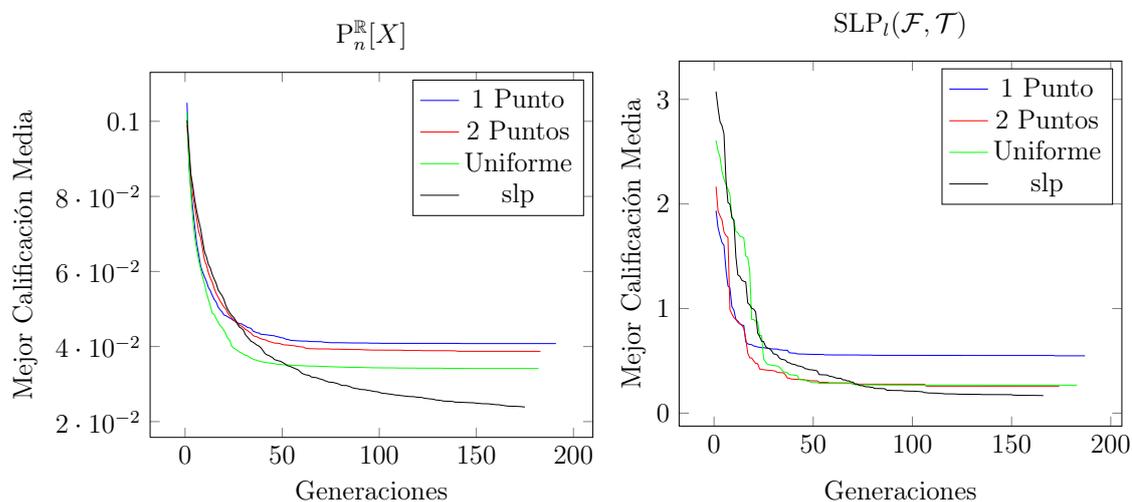
$P_n^R[X]$	1 Punto	2 Puntos	Uniforme	slp
1 Punto	1	0,85	1	1
2 Puntos	0,43	1	1	1
Uniforme	$4,23 \cdot 10^{-2}$	0,11	1	1
slp	$2,26 \cdot 10^{-9}$	$1,07 \cdot 10^{-8}$	$2,15 \cdot 10^{-5}$	1

$SLP_l(\mathcal{F}, \mathcal{T})$	1 Punto	2 Puntos	Uniforme	slp
1 Punto	1	0,91	1	1
2 Puntos	0,12	1	1	1
Uniforme	$2,19 \cdot 10^{-2}$	$7,14 \cdot 10^{-2}$	1	1
slp	$3,81 \cdot 10^{-6}$	$7,99 \cdot 10^{-5}$	$1,34 \cdot 10^{-2}$	1

Ya hemos aventurado a lo largo de la Sección 8.2.3 la causa que explica este comportamiento pero en la Figura 8.9 se observa con claridad. En la siguiente figura representamos

como evoluciona la calificación media del mejor elemento con respecto a la generación en la que se encuentra.

Figura 8.9: Evolución de la Calificación: *cruces para slp*.



Como se puede observar claramente, las calificaciones medias disminuyen rápidamente al principio para luego «estancarse». En los cruces en un punto, en dos puntos y uniforme la disminución es muy pronunciada al principio pero el estancamiento sucede muchas generaciones antes que en el cruce *slp*. Este fenómeno se suele denominar *convergencia prematura* y es uno de los fenómenos a evitar. Generalmente se da porque la población pierde diversidad puesto que alcanza un mínimo local que rápidamente se propaga por toda la población en virtud del Teorema de los Esquemas (véase Teorema 0.0.39).

8.3.2 Árboles frente a *slp*'s

Los parámetros específicos de este experimento son:

Cuadro 8.6: Parámetros específicos: *slp frente a árbol*.

Tipo de estrategia:	Sólo programación genética
Estructura de datos:	{ <i>slp</i> , árbol}
Operador de calificación:	<i>MSE</i>
Operador de cruce:	{ <i>slp</i> , árbol}
Operador de mutación:	{ <i>slp</i> , árbol}

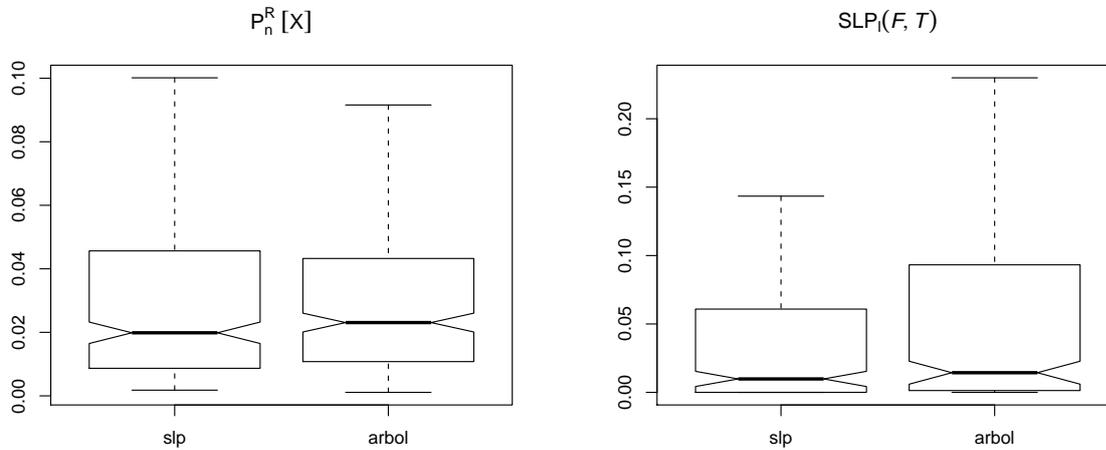
En la Tabla 8.7 volvemos a observar el mismo patrón para la tasa de error como para la tasa de éxito: son muy similares entre ambas estructuras de datos y en el caso de las muestras generadas sobre *slp*'s aleatorios peores que sobre las muestras generadas mediante polinomios.

Cuadro 8.7: Tasas de error y de solución: *slp* frente a *arbol*.

$P_n^{\mathbb{R}}[X]$	t_e	t_s	$SLP_l(\mathcal{F}, \mathcal{T})$	t_e	t_s
slp	8,33 %	100 %	slp	15,67 %	97,67 %
arbol	5,67 %	100 %	arbol	14,33 %	98,33 %

De la misma forma, observamos en los los diagramas de cajas de la Figura 8.10 diferencias significativas entre usar *slp*'s a usar árboles. Esto último se observa sobre todo en las muestras generadas sobre *slp*'s aleatorios, nótese que aunque la caja es más ancha en los resultados sobre las muestras generadas con polinomios aleatorios, la mediana esta sensiblemente más abajo.

Figura 8.10: Diagramas de Cajas y Bigotes: *slp* frente a *arbol*.



Las Tablas 8.8 y 8.9 no introducen ninguna diferencia significativa respecto a lo anteriormente mostrado.

Cuadro 8.8: Resumen estadístico I: *slp* frente a *arbol*.

$P_n^{\mathbb{R}}[X]$	μ	σ	$SLP_l(\mathcal{F}, \mathcal{T})$	μ	σ
slp	$2,55 \cdot 10^{-2}$	$2,3 \cdot 10^{-2}$	slp	$2,05 \cdot 10^{-2}$	$3,22 \cdot 10^{-2}$
arbol	$2,69 \cdot 10^{-2}$	$2,07 \cdot 10^{-2}$	arbol	$3,17 \cdot 10^{-2}$	$4,96 \cdot 10^{-2}$

Cuadro 8.9: Resumen estadístico II: *slp* frente a *arbol*.

$P_n^{\mathbb{R}}[X]$	mín	q_1	q_2	q_3	máx
slp	$1,79 \cdot 10^{-3}$	$8,27 \cdot 10^{-3}$	$1,8 \cdot 10^{-2}$	$3,62 \cdot 10^{-2}$	0,1
arbol	$1,1 \cdot 10^{-3}$	$1,02 \cdot 10^{-2}$	$2,18 \cdot 10^{-2}$	$3,74 \cdot 10^{-2}$	$9,16 \cdot 10^{-2}$

$SLP_l(\mathcal{F}, \mathcal{T})$	mín	q_1	q_2	q_3	máx
slp	0	$3,08 \cdot 10^{-33}$	$4,13 \cdot 10^{-3}$	$2,76 \cdot 10^{-2}$	0,14
arbol	0	$6,54 \cdot 10^{-4}$	$9,52 \cdot 10^{-3}$	$3,57 \cdot 10^{-2}$	0,22

Los contrastes de hipótesis no dejan lugar a dudas. Para cualquier valor razonable de la significancia α , usar la estructura de datos *slp* es significativamente superior a la estructura de datos árbol.

Cuadro 8.10: Contraste de Hipótesis: *slp* frente a árbol.

$P_n^{\mathbb{R}}[X]$	slp	arbol	$SLP_t(\mathcal{F}, \mathcal{T})$	slp	arbol
slp	1	$8,07 \cdot 10^{-2}$	slp	1	$7,16 \cdot 10^{-3}$
arbol	0,79	1	arbol	1	1

Capítulo 9

La dimensión de Vapnik-Chervonenkis de un *slp*

*Pluralitas non est ponenda sine necessitate*¹.
William of Ockham [oO95, i, dist. 27, qu. 2, K]

A lo largo de estas páginas presentaremos un estudio teórico de las propiedades de los *slp*'s. En particular, nos centraremos en calcular una cota superior de la dimensión de Vapnik-Chervonenkis (VCD) de ciertas clases de *slp*'s (véase [MABC09]).

Teorema 9.0.1. *Sea $\text{SLP}_{n,L}(\mathcal{F}, \mathcal{T})$ el conjunto de todos los *slp*'s sobre el conjunto de terminales $\mathcal{T} = \{t_1, \dots, t_n\}$ y el conjunto de funciones $\mathcal{F} = \{+, -, *, /, \text{sgn}, f_1, \dots, f_q\}$ donde $\{+, -, *, /\}$ denotan las funciones aritméticas estándar, $\text{sgn}(x)$ es la función:*

$$\begin{aligned} \text{sgn} : \mathbb{R} &\longrightarrow \mathbb{R} \\ x &\longmapsto \text{sgn}(x) = \begin{cases} 0 & \text{si } x \leq 0. \\ 1 & \text{si } x > 0. \end{cases} \end{aligned}$$

y f_1, \dots, f_q es una cadena de Pfaffian de grado D (véase la Definición 9.1.4). Además, los *slp*'s de dicho conjunto cumplen las siguientes restricciones:

- El número de operaciones no escalares (i.e. operaciones en $\{*, /, \text{sgn}, f_1, \dots, f_q\}$) esta acotado por L .
- El número de operaciones escalares (i.e. operaciones en $\{+, -\}$) no está restringido.
- El número total de parámetros es k .

Sea ahora $\mathcal{C}_{n,L}$ la clase de conceptos definidos por los subconjuntos de \mathbb{R}^n que son aceptados por un *slp* (véase la Definición 9.1.2) en el conjunto $\text{SLP}_{n,L}(\mathcal{F}, \mathcal{T})$. Entonces, se tiene que:

$$\begin{aligned} VC_{\dim}(\mathcal{C}_{n,L}) &\leq \\ &\leq \mathcal{O}((q(N+k)(L+1))^2 + q(N+k)(L+1) \log_2((N+k)(L+1)(4+D))) \end{aligned}$$

donde $N = \mathcal{O}(L^2A + LA(n+k) + Lq)$.

¹La pluralidad no se debe postular sin necesidad.

Posteriormente, a partir de dicha cota, construiremos un operador de calificación que permitirá realizar minimización del riesgo estructural (SRM) cuando se evolucionan slp 's.

Finalmente, realizaremos un estudio empírico, siguiendo el protocolo del Anexo B, en el que compararemos el comportamiento de este nuevo procedimiento calificador con otros procedimientos SRM clásicos adaptados a la programación genética en el que se observa que el procedimiento calificador basado en la VCD es superior de forma estadísticamente significativa (véase [MABC09] y [BAM10] para otros resultados en este sentido).

9.1 Estimación de la VCD de familias de slp 's parametrizadas por números reales

En la década de los setenta, los trabajos de Vapnik y Chervonenkis (véase [VC74, Vap98, VC71]) proporcionaron una familia muy importante de cotas relacionadas con la capacidad de clasificación de una *máquina de aprendizaje automático* (véase [Lug02] para una discusión en términos modernos de la teoría). La VCD mide la capacidad de una familia de funciones (o máquinas de aprendizaje automático) como clasificadores. Ya que la VCD depende de la clase de clasificadores no tiene sentido calcular la VCD de la programación genética en general aunque si es interesante realizar el estudio para el conjunto particulares como, por ejemplo, el de los slp 's.

La siguiente es la definición formal de VCD. Véase [Vap98] para seguir los detalles.

Definición 9.1.1 (VCD). *Sea \mathcal{C} una clase de subconjuntos de un conjunto X . Se dice que \mathcal{C} ajusta el conjunto $A \subset X$ si para todo subconjunto $E \subset A$ existe un elemento $S \in \mathcal{C}$ tal que $E = S \cap A$. La VCD del conjunto \mathcal{C} será el cardinal del mayor conjunto A que es ajustado por \mathcal{C} .*

A lo largo de este capítulo usaremos la notación $\mathcal{C}_{k,n}$ para referirse a un conjunto de conceptos representados por k números reales, $w = (w_1, \dots, w_k)$, las muestras estarán representadas por n números reales, $x = (x_1, \dots, x_n)$, y el test de pertenencia a $\mathcal{C}_{k,n}$ es expresado mediante una fórmula $\Phi_{k,n}(w, x)$ que tiene como parámetros de entrada pares (w, x) donde w representa un concepto y x una muestra. Esta fórmula devuelve un valor 1 en el caso que « x pertenezca al concepto representado por w » y 0 en otro caso. De esta forma, podemos pensar en $\Phi_{k,n}$ como en una aplicación de \mathbb{R}^{k+n} al conjunto $\{0, 1\}$.

Definición 9.1.2. *Diremos que un subconjunto $C \subset \mathbb{R}^n$ es aceptado por un slp Γ si la aplicación semántica Φ_Γ expresa la función de pertenencia de C .*

Para cada concepto w se define por C_w el conjunto de todas las instancias aceptadas por el concepto w . Es decir,

$$C_w := \{x \in \mathbb{R}^n : \Phi_{k,n}(w, x) = 1\}.$$

El objetivo es calcular una cota superior para la VCD del conjunto

$$\mathcal{C}_{k,n} = \{C_w : w \in \mathbb{R}^k\}$$

asumiendo que la fórmula $\Phi_{k,n}$ es una combinación booleana de s fórmulas atómicas de la forma:

$$\tau_i(w, x) > 0 \text{ o bien } \tau_i(w, x) = 0,$$

donde $\{\tau_i(w, x)\}_{1 \leq i \leq s}$ son funciones infinitamente diferenciables de \mathbb{R}^{k+n} a \mathbb{R} .

Sea $\alpha_1, \dots, \alpha_v \in \mathbb{R}^n$ un conjunto de muestras. De entre las $s \cdot v$ funciones $\tau_i(w, \alpha_j)$ de \mathbb{R}^k a \mathbb{R} tomamos r , $\Theta_1, \dots, \Theta_r$, y definimos

$$\begin{aligned} \Theta : \mathbb{R}^k &\longrightarrow \mathbb{R}^r \\ w &\longmapsto (\Theta_1(w), \dots, \Theta_r(w)). \end{aligned}$$

El siguiente resultado es debido a Karpinski ([KM97]) y permite acotar la VCD a partir de invariantes geométricos de los test de pertenencia:

Teorema 9.1.3. *Usando la notación anterior. Sea $(\epsilon_1, \dots, \epsilon_r) \in \mathbb{R}^r$ un punto tal que $\Theta^{-1}(\epsilon_1, \dots, \epsilon_r)$ es un subvariedad real \mathcal{C}^∞ de dimensión $(k-r)$. Si existe una cota superior B al número de componentes conexas de $\Theta^{-1}(\epsilon_1, \dots, \epsilon_r)$ que: ni dependa del número r ni de la muestra α_i escogida se tiene que, la VCD de una familia de conceptos $\mathcal{C}_{k,n}$ cuya función de pertenencia se puede expresar mediante una fórmula $\Phi_{k,n}$ que satisface las condiciones anteriores cumple:*

$$VC_{dim}(\mathcal{C}_{k,n}) \leq 2 \log_2 B + 2k \log_2(2es).$$

Pasamos a demostrar el Teorema Principal de este capítulo. Antes debemos introducir algunas nociones sobre las cadenas de Pfaffian y sobre sus propiedades geométricas. En particular introducimos un lema de J. Milnor (véase [Mil64]) ampliado posteriormente por Gabrielov y Vorobjov (véase [GV04]).

Definición 9.1.4 (Cadena de Pfaffian). *Sea $U \subset \mathbb{R}^n$ un dominio abierto. Una cadena de Pfaffian de longitud $q \geq 0$ y grado $D \geq 1$ en U es una secuencia de funciones reales analíticas f_1, \dots, f_q que satisfacen las siguientes ecuaciones diferenciales:*

$$\frac{\partial f_i}{\partial x_j} = P_{i,j}(x, f_1(x), \dots, f_i(x)),$$

con $1 \leq i \leq q$ y donde $P_{i,j} \in \mathbb{R}[x_1, \dots, x_n, y_1, \dots, y_i]$ son polinomios de grado menor o igual que D .

Una función f definida en U se dice de Pfaffian de orden q y grados (d, D) si

$$f(x) := P(x, f_1(x), \dots, f_q(x))$$

con $P \in \mathbb{R}[x_1, \dots, x_n, y_1, \dots, y_i]$ de grado menor que d y $f_1(x), \dots, f_q(x)$ forman una cadena de Pfaffian de grado D .

Entre los ejemplos de estas funciones podríamos citar:

- Todas las funciones algebraicas en sus respectivos dominios.

- $\log x$ y e^x así como las funciones hiperbólicas igualmente en los dominios en los que están definidas.
- Las funciones trigonométricas en intervalos acotados.
- En general todas las llamadas funciones elementales en sus respectivos dominios son Pfaffian.

Lema 9.1.5 (Milnor [Mil64], Gabrielov y Vorobjov [GV04]). *Usando la notación anterior y asumiendo que $\Theta_1(w), \dots, \Theta_r(w)$ son polinomios con k variables y grado acotado por d . Entonces, si $\epsilon = (\epsilon_1, \dots, \epsilon_r)$ es un valor regular de $\Theta(w) := (\Theta_1(w), \dots, \Theta_r(w))$, se tiene que el número de componentes conexas B está acotado por:*

$$B \leq (2d)^k. \quad (9.1)$$

En el caso que $\Theta_1(w), \dots, \Theta_r(w)$ sean polinomios en el anillo $\mathbb{R}[y_1, \dots, y_k, f_1, \dots, f_q]$ de grado acotado por d donde f_1, \dots, f_q forman una cadena de Pfaffian de longitud q y grado D la anterior cota se transforma en:

$$B \leq 2^{q(q-1)/2} d^k (k(d+D))^k (k^2(d+D))^q. \quad (9.2)$$

Nota 9.1.6. *Nótese que la anterior cota quedaría*

$$B \leq 2^{kq(kq-1)/2} d^k (k(d+D))^k (k^2(d+D))^{kq}$$

al sustituir x por α_i , $1 \leq i \leq r$ en la cadena de Pfaffian inicial f_1, \dots, f_q . Por lo tanto tenemos como únicas variables y_1, \dots, y_k y una cadena de Pfaffian de longitud $rq \leq kq$.

Antes de afrontar la demostración del teorema principal de esta sección necesitamos introducir un lema en el que determinaremos el número de parámetros necesarios para construir una parametrización del conjunto de las aplicaciones semánticas pertenecientes a la familia de slp 's definidos sobre los conjuntos \mathcal{F} y \mathcal{T} . Denominaremos a dicha parametrización slp universal.

Lema 9.1.7. *Sea $\mathcal{T} := \{t_1, \dots, t_n\}$ un conjunto de terminales y $\mathcal{F} := \{+, -, *, /, \text{sgn}\} \cup \{f_1, \dots, f_q\}$ el conjunto de funciones donde los elementos $\{f_1, \dots, f_q\}$ forman una cadena de Pfaffian de longitud q y grado D . Además supondremos que la aridad de dichas funciones está acotada por A . Entonces, el número de parámetros del slp universal Γ_u para la familia de slp 's $\text{SLP}_{n,L}(\mathcal{F}, \mathcal{T})$ es:*

$$L \left(3 + q + A \left(n + \frac{L-1}{2} \right) + 1 \right) + n.$$

Demostración. Introduciremos un conjunto de parámetros que denotaremos con los símbolos α , β y γ pertenecientes a \mathbb{Z}^a para un determinado a . El objetivo es encontrar una expresión $\Gamma_U(\alpha, \beta, \gamma)$ de forma que para todo slp del conjunto $\text{SLP}_{n,L}(\mathcal{F}, \mathcal{T})$ su aplicación semántica queda determinada por unos valores particulares de α , β y γ . Para simplificar la notación, renombramos los elementos del conjunto de terminales $u_{-n+m} = t_m$, para todo $1 \leq m \leq n$.

Sea Γ un slp perteneciente a la familia de slp 's $SLP_{n,L}(\mathcal{F}, \mathcal{T})$. Claramente se observa que toda variable no terminal con operación no escalar u_i , con $i \leq i \leq L$, de Γ es una función que se puede parametrizar de la siguiente forma:

$$u_i := U_i(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})(\mathbf{t}) = \gamma_{-n}^i \left[\alpha_{-n}^i \left(\sum_{j=-n+1}^{i-1} \alpha_j^{i_1} u_j \right) \left(\sum_{j=-n+1}^{i-1} \alpha_j^{i_2} u_j \right) + \right. \\ \left. + (1 - \alpha_{-n}^i) \left[\beta_{-n}^i \frac{\sum_{j=-n+1}^{i-1} \alpha_j^{i_1} u_j}{\sum_{j=-n+1}^{i-1} \alpha_j^{i_2} u_j} + (1 - \beta_{-n}^i) \text{sgn} \left(\sum_{j=-n+1}^{i-1} \alpha_j^{i_1} u_j \right) \right] + \right. \\ \left. + (1 - \gamma_{-n}^i) \sum_{k=1}^q \gamma_k^i f_k \left(\sum_{j=-n+1}^{i-1} \alpha_j^{i_1} u_j, \dots, \sum_{j=-n+1}^{i-1} \alpha_j^{i_A} u_j \right) \right].$$

Para terminar consideramos, sin pérdida de generalidad, que la última variable no terminal se puede parametrizar de la siguiente manera:

$$U = \sum_{j=-n+1}^L \alpha_j u_j,$$

donde u_j representa las L variables no terminales con operaciones no escalares. Esto se debe a que tras la última operación no escalar puede existir un número arbitrario de operaciones escalares.

Nótese que los parámetros α_{-n}^i , β_{-n}^i y γ_{-n}^i , toman valores en el conjunto $\{0, 1\}$ mientras que el resto lo hace en \mathbb{Z} . Esto se debe a que en cada variable no terminal solo se realiza una de las operaciones no escalares. De esta forma obtenemos una estructura universal

$$\Gamma_U = \{U_1(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}), \dots, U_L(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}), U(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})\}$$

que parametriza cualquier aplicación semántica de la familia de slp 's. La ecuación del enunciado se obtiene inmediatamente contando el número de parámetros $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ y $\boldsymbol{\gamma}$. \square

Procedemos ahora a la demostración del teorema principal de este capítulo:

9.1.1 Demostración del Teorema 9.0.1

Sea $SLP_{n+k,L}(\mathcal{F}, \mathcal{T})$ la familia de slp 's definidos sobre los conjuntos $\mathcal{T} = V \cup C$ donde $V = \{x_1, \dots, x_n, y_1, \dots, y_k\}$ representa el par (instancia, concepto) y $\mathcal{F} = \{+, -, *, /, \text{sgn}\} \cup \{f_1, \dots, f_q\}$ con f_1, \dots, f_q denotando una cadena de Pfaffian de longitud q y grado D . Además suponemos que los slp 's pretenecientes a dicha familia tendrá a lo sumo L operaciones no escalares y un número finito, pero arbitrario, de operaciones escalares.

Aplicando el Lema 9.1.7 sabemos que existe un slp universal

$$\Gamma_U = \{U_1(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}), \dots, U_L(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}), U(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})\}$$

que parametriza todas las aplicaciones semánticas de todos los elementos de la familia. De esta forma podemos reemplazar la familia de conceptos $\mathcal{C}_{k,n}$ por la familia de subconjuntos de \mathbb{R}^{n+k}

$$\mathcal{C} := \{C_{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}}\}$$

donde para cada $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ el conjunto $C_{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}}$ esta definido como:

$$C_{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}} := \{(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{n+k} : (\boldsymbol{x}, \boldsymbol{y}) \text{ es aceptado por } \Gamma_{U_{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}}}\}.$$

Denotaremos por $N := N(L, k + n, q, A)$ al número de parámetros que posee. Incorporamos al conjunto de variables dichos parámetros. De esta manera tenemos $n + k + N$ variables. Como anteriormente hemos hecho, renombramos el conjunto de terminales \mathcal{T} para facilitar la demostración. De esta forma, tomamos

$$\mathcal{T} = \{u_{-m+1}, \dots, u_0\},$$

donde $m = k + n$. De esta forma las operaciones que se pueden dar en Γ_U pueden ser de alguna de las siguientes formas:

1. $u_i := \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} u_j \right) \text{ op } \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_2} u_j \right)$ con $op \in \{*, /\}$.
2. $u_i := \sum_{k=1}^q \gamma_k^i f_k \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} u_j, \dots, \sum_{j=-m+1}^{i-1} \alpha_j^{i_A} u_j \right)$.
3. $u_i := \text{sgn} \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} u_j \right)$.

Claramente se tiene que Γ_U tiene a lo sumo L operaciones sgn . Para cada una de ellas definimos

$$U_i := \{u_{i_1}, \dots, u_{i_k}\}$$

como el conjunto de operaciones sgn computadas antes que u_i . Luego, en la computación de la variable no terminal u_i , las variables en el conjunto U_i toman valores fijos en el conjunto $\{0, 1\}$ por lo que podemos considerarla como una fórmula

$$\Theta_i(u_{-m+1}, \dots, u_{i-1}) = \Theta_i(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{x}, \boldsymbol{y}, u_{-m+1}, \dots, u_{i-1}),$$

donde las instrucciones signo anteriores verifican que

$$u_{i_s} = \delta_s \in \{0, 1\} \text{ con } u_{i_s} \in U_i.$$

Tenemos entonces que Θ_i es un predicado atómico $\tau(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{x}, \boldsymbol{y})$ para cada $(\delta_1, \dots, \delta_{k_i}) \in \{0, 1\}^{k_i}$. Definimos ahora para cada instrucción de signo u_i el siguiente predicado $W(u_i)$:

$$W(u_i) := \{\tau(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{x}, \boldsymbol{y}) : (\delta_1, \dots, \delta_{k_i}) \in \{0, 1\}^{k_i}\}. \quad (9.3)$$

Los predicados $W(u_i)$ pertenecen a un lenguaje de primer orden L sobre los números reales con símbolos $+, -, *, /, f_1, \dots, f_q$ y el orden $<$ donde f_1, \dots, f_q es una cadena de Pfaffian. Como $k_i < i$ se tiene que $W(u_i)$ contiene a lo sumo $2 \cdot 2^{i-1} = 2^i$ predicados. Por lo tanto, la fórmula que evalúa el test de pertenencia $\Gamma_{U_{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}}}$ consiste en a lo sumo $L \cdot 2^L$ predicados.

Sea $W := \bigcup_{i=1}^L W(u_i)$. De acuerdo con el Teorema 9.1.3 debemos acotar el número de componentes conexas de la variedad $(k + N - r)$ -dimensional definida por:

$$\tau_{\alpha_i}(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{y}) - \epsilon_i = 0,$$

donde $1 \leq i \leq r \leq k + N$, $\alpha_i \in \mathbb{R}^n$ y $\tau_{\alpha_i} := \tau(\alpha, \beta, \gamma, \alpha_i, \mathbf{y})$ pertenece a W .

Asumamos que τ_{α_i} pertenece al conjunto $W(u_i)$ y sea $(\delta_1, \dots, \delta_{k_i}) \in \{0, 1\}^{k_i}$ los valores asociados de acuerdo a la Ecuación (9.3). Usamos ahora las cotas ofrecidas por la Ecuación (9.2) del Lema 9.1.5. Crearemos una variedad con las siguientes ecuaciones:

Para cada variable no terminal $u_k \in \Gamma_U$, con $1 \leq k \leq i - 1$ introducimos en el sistema las ecuaciones:

1. Si $u_k := \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} u_j \right) \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_2} u_j \right)$, entonces introducimos la variable v_k junto con la ecuación:

$$v_k - \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} v_j \right) \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_2} v_j \right) = 0. \quad (9.4)$$

2. Si $u_k := \frac{\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} u_j}{\sum_{j=-m+1}^{i-1} \alpha_j^{i_2} u_j}$, entonces introducimos la variable v_k junto con la ecuación:

$$v_k \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} v_j \right) - \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_2} v_j \right) = 0. \quad (9.5)$$

3. Si $u_k := \sum_{l=1}^q \gamma_l^i f_l \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} u_j, \dots, \sum_{j=-m+1}^{i-1} \alpha_j^{i_A} u_j \right)$, entonces introducimos la variable v_k junto con la ecuación:

$$v_k - \sum_{l=1}^q \gamma_l^i f_l \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} v_j, \dots, \sum_{j=-m+1}^{i-1} \alpha_j^{i_A} v_j \right) = 0. \quad (9.6)$$

4. Si $u_k := \operatorname{sgn} \left(\sum_{j=-m+1}^{i-1} \alpha_j^{i_1} u_j \right)$, entonces introducimos la variable v_k junto con la ecuación:

$$v_k - \delta_k = 0, \quad \text{donde } \delta_k \in \{0, 1\}. \quad (9.7)$$

Finalmente añadimos la ecuación

$$v_i - \epsilon_i = 0. \quad (9.8)$$

Sea $S(\alpha, \beta, \gamma, \mathbf{y}, \epsilon_1, \dots, \epsilon_r, \mathbf{v})$ el sistema obtenido a partir del conjunto de Ecuaciones (9.4), (9.5), (9.6), (9.7) y (9.8) para todo i tal que $1 \leq i \leq r$, donde hemos denotado por v las nuevas variables. Tenemos entonces que:

1. $S(\alpha, \beta, \gamma, \mathbf{y}, \epsilon_1, \dots, \epsilon_r, \mathbf{v}) \Rightarrow \tau_{\alpha_i}(\alpha, \beta, \gamma, \mathbf{y}) = \epsilon_i$ para todo i en $\{1, \dots, r\}$.
2. Si $\tau_{\alpha_i}(\alpha, \beta, \gamma, \mathbf{y}) = \epsilon_i$ para todo i en $\{1, \dots, r\}$ entonces existen unos únicos v tales que se cumple el sistema de ecuaciones $S(\alpha, \beta, \gamma, \mathbf{y}, \epsilon_1, \dots, \epsilon_r, \mathbf{v})$.
3. El conjunto definido por las variables \mathbf{y} mediante las ecuaciones $\tau_{\alpha_i}(\alpha, \beta, \gamma, \mathbf{y}) = \epsilon_i$ para todo i en $\{1, \dots, r\}$ es homeomorfo al conjunto definido mediante las variables $\alpha, \beta, \gamma, \mathbf{y}$ y \mathbf{v} en el sistema $S(\alpha, \beta, \gamma, \mathbf{y}, \epsilon_1, \dots, \epsilon_r, \mathbf{v})$. Luego o ambos son una variedad o ninguno lo es.

Hemos construido un sistema S que contiene a lo sumo $r(L + 1) \leq (k + N)(L + 1)$ ecuaciones. Dichas ecuaciones son polinomios de grado acotado por 4 en el anillo de polinomios $\mathbb{R}[\alpha, \beta, \gamma, \mathbf{y}, \mathbf{v}, \mathbf{f}]$ donde f es una cadena de Pfaffian de longitud $rq \leq (k + N)q$ por la misma razón que la descrita en la Nota 9.1.6. El número de variables esta acotado por $N + k + rL \leq (k + N)(1 + L)$. Asumiendo que el sistema S define una variedad y usando las cotas dadas por la Ecuación (9.2) del Lema 9.1.5 obtenemos que el número de componente conexa B está acotado por:

$$B \leq 2^{q(N+k)(L+1)(q(N+k)(L+1)-1)/2} 4^{(N+k)(L+1)} \\ ((k + N)(L + 1)(4 + D))^{(k+N)(L+1)} ((N + k)^2(L + 1)^2(4 + D))^{q(N+k)(L+1)}$$

donde

$$N = L \left(3 + q + A \left(n + k + \frac{L - 1}{2} \right) + 1 \right) + n + k.$$

El resultado final se obtiene usando la anterior cota en el Teorema 9.0.1 y tomando $s = L \cdot 2^L$ como cota del número de predicados. \square

Nota 9.1.8. Si el conjunto de funciones de nuestro slp solo contiene las operaciones aritméticas elementales, es decir $\mathcal{F} = \{+, -, *\}$ entonces podemos usar la cota dada por la Ecuación (9.1) del Lema 9.1.5 y la cota para la VCD quedaría:

$$VC_{dim}(\mathcal{C}_{n,L}) \leq 2(n + 1)(n + L)L(2L + \log_2 L + 9).$$

9.2 Calificación basada en la VCD de un slp

A lo largo de esta sección introduciremos diversas formas de escoger una calificación para estimar el error medio cometido respecto de la distribución de probabilidad (que recordemos es desconocida) desde la cual se presentan las muestras.

Comenzaremos proponiendo el uso de las herramientas que nos ofrece la *Teoría del Aprendizaje Estadístico* (véase [Vap98, VC71, VC74],[CY03]). Este punto de vista no es nuevo y ya ha sido ensayado en [TGBS05]. Presentaremos dos enfoques para atacar este problema. Por un lado adaptaremos algunos métodos estadísticos clásicos como criterio de información de Akaike (AIC) y criterio de información Bayesiano (BIC) al contexto de la programación genética mientras que por otro lado usaremos el método de minimización del riesgo estructural (SRM). El objetivo es, recordemos, estimar una función g tal que:

$$y = g(x) + \epsilon \tag{9.9}$$

donde ϵ denota una fuente aleatoria de errores con media 0 e i.i.d. (comúnmente se denomina a esta fuente de errores como *ruido*), x es el vector de entrada e y el valor de salida. Para realizar la estimación contamos con una muestra basada en un número finito n de muestras que denominaremos muestra de aprendizaje. Esta muestra de aprendizaje estará formada por pares $(x_i, y_i)_{1 \leq i \leq n}$ i.i.d. generados de acuerdo a una función de densidad conjunta desconocida:

$$\rho(x, y) = \rho(x)\rho(y|x).$$

Es bien conocido que el mejor estimador f_{best} para la función g en la Ecuación (9.9) que se puede obtener a partir de la muestra de aprendizaje es la media respecto de la distribución de probabilidad condicionada. Es decir:

$$f_{best}(x) = \int y\rho(y|x).$$

Desgraciadamente, como ya hemos mencionado, ρ suele ser desconocida y por lo tanto dicha integral es imposible de calcular (ni siquiera aproximadamente).

Una máquina de aprendizaje automático tratará de seleccionar el mejor modelo $f \in \mathcal{H}$, donde \mathcal{H} es una clase de conceptos. En general, el error del estimador f , se denota por $\varepsilon(f)$ y cumple:

$$\varepsilon(f) = \int Q(f, x, y)d\rho, \quad (9.10)$$

donde Q mide la pérdida de información de usar el modelo $f(x)$ respecto del concepto g y ρ es la distribución respecto de la que la muestra de aprendizaje (x, y) ha sido creada. Para problemas de clasificación, el error de clasificación Q que se suele usar es el número de muestras mal clasificadas. Esto es:

$$Q(f, x, y) = |y - f(x)|.$$

Mientras que, para problemas de regresión simbólica, se suele usar el llamado *MSE*. Es decir:

$$Q(f, x, y) = (y - f(x))^2.$$

Puesto que ρ es desconocido, los errores anteriormente descritos no son posibles de calcular. En el caso de que la clase de modelos \mathcal{H} tenga complejidad finita (en nuestro caso, cuando acotamos la longitud de los *slp's* o la altura de los árboles), podemos usar como error del modelo el *error empírico* (también llamado *riesgo empírico*) en vez del *error teórico* definido en la Ecuación (9.10):

$$\varepsilon_n(f) = \frac{1}{n} \sum_{i=1}^n Q(f, x_i, y_i) \quad (9.11)$$

El problema de la *selección del modelo* (también llamada *control de la complejidad*) aparece cuando los modelos presentan distinta complejidad, como es nuestro caso. Cuando se trata de problemas de regresión simbólica se trata de escoger el modelo de menor complejidad que minimice el riesgo, es decir, minimiza el valor de la función definida en la Ecuación (9.11). Para ello, usualmente, se estima el error teórico (Ecuación (9.10)) mediante una función del error empírico (Ecuación (9.11)) añadiéndole cierta penalización relacionada de alguna manera con la complejidad del modelo. Es esta función la que luego se usa para estimar el mejor modelo en la clase \mathcal{H} . A esta forma de seleccionar el modelo se la conoce comúnmente como minimización del riesgo estructural (*SRM*).

La *SRM* da buenos resultados suponiendo que los modelos cumplen una serie de propiedades (véase [CY03] y referencias): principalmente linealidad y exactitud en el cálculo de la capacidad de clasificación de la clase de conceptos \mathcal{H} . Como en el caso de la programación genética ninguna de las dos condiciones se cumple, es necesario realizar un estudio empírico para comprobar si los resultados teóricos siguen siendo válidos en este contexto.

9.3 Criterios para la selección del modelo

Como mencionamos anteriormente, en general, para estimar el error teórico (Ecuación (9.10)) se suele penalizar el error empírico (Ecuación (9.11)) de alguna de las siguientes formas:

$$\begin{aligned}\varepsilon(f) &= \varepsilon_n(f) \cdot \text{pen}(h, n), \\ \varepsilon(f) &= \varepsilon_n(f) + \text{pen}(h, n, \sigma^2),\end{aligned}$$

donde f representa el modelo, pen es el factor de penalización, h es la complejidad del modelo, n es el tamaño de la muestra de aprendizaje y finalmente σ denota la desviación típica del ruido (véase la Ecuación (9.9)).

Los dos métodos estadísticos que usaremos en este trabajo son bien conocidos:

- *AIC* (véase [Aka70]):

$$\varepsilon(f) = \varepsilon_n(f) + \frac{2h}{n}\sigma^2.$$

- *BIC* (véase [BS94]):

$$\varepsilon(f) = \varepsilon_n(f) + (\ln n)\frac{h}{n}\sigma^2.$$

Como se describe en [CY03], cuando se usa un estimador lineal f , primero se estima la varianza del ruido que presenta la muestra (x_i, y_i) como:

$$\sigma^2 = \frac{n}{n-h} \frac{1}{n} \sum_{1 \leq i \leq n} (y_i - f(x_i))^2 \quad (9.12)$$

donde $f(x_i)$ es el valor que el estimador lineal f propone para el valor y_i . Posteriormente se usa la Ecuación (9.12) junto con *AIC* o *BIC* para estimar la complejidad del modelo. En ambos casos se usará como estimador de la complejidad del modelo f el número de parámetros libres del estimador f .

El tercer mecanismo de selección del modelo que usaremos se basa en el método *SRM* (véase [Vap98])

$$\varepsilon(f) \leq \varepsilon_n(f) + \sqrt{\frac{h(\log(2n/h) + 1) - \log(\eta/4)}{n}}, \quad (9.13)$$

donde h debe ser sustituida por una cota superior de la VCD como la conseguida en el Teorema 9.0.1. Por otro lado, la constante η representa la probabilidad de que esta cota sea violada.

Bajo las hipótesis de *SRM* un conjunto de posibles modelos C forman una estructura anidada

$$C_1 \subset C_2 \subset \dots \subset C_L \subset \dots \subset C,$$

donde C_L representa el el conjunto de modelos de complejidad L con L un parámetro que depende del problema. Se requiere que la VCD de los modelos $VC_{\dim}(C_L)$ sea una función creciente sobre L .

En este caso particular, C es la clase de slp 's de longitud acotada por l sobre el conjunto de terminales $\mathcal{T} = \{t_1, \dots, t_n\}$ y el conjunto de funciones $\mathcal{F} = \{+, -, *, /, \text{sgn}, f_1, \dots, f_q\}$ y C_L es la clase de slp 's en C que usan como mucho L operaciones no escalares. Bajo estas hipótesis, siguiendo el principio de la *navaja de Ockham*, se trata de escoger el modelo que minimice la parte derecha de la Ecuación (9.13).

En la práctica se suele escoger la siguiente fórmula en vez de la provista por la Ecuación (9.13) ajustando convenientemente los valores (véase [CY03] si se está interesado en los detalles de cómo se deriva esta fórmula):

$$\varepsilon(f) = \varepsilon_n(f) \left(1 - \sqrt{\frac{h}{n} \left[1 - \ln \left(\frac{h}{n} \right) \right] + \frac{\ln n}{2n}} \right)^{-1},$$

donde h denota la VCD del modelo. Nótese que bajo esta aproximación no es necesario estimar la varianza del ruido pero si es necesario contar con una estimación de la VCD.

En la Sección 9.4 presentamos un estudio empírico del desempeño de estos procedimientos calificadores en el que se observa como el método basado en la *SRM* es superior de forma estadísticamente significativa (véase [MABC09] y [BAM10] para otros resultados en este sentido).

9.4 Resultados experimentales

En las siguientes páginas realizaremos un estudio empírico en el que mostraremos que en problemas de regresión simbólica con muestras que contienen ruido la calificación basada en *SRM* produce resultados estadísticamente significativos a su favor cuando se la compara con los métodos estadísticos *AIC* y *BIC* y, por supuesto, cuando se la compara con la calificación basada en el *MSE*. Para llevar a cabo dicho estudio seguiremos el protocolo propuesto en el Anexo B. Pasamos ahora a detallar los parámetros particulares de este experimento.

Los parámetros específicos de este experimento son:

Cuadro 9.1: Parámetros específicos: *métodos de calificación con ruido*.

Tipo de estrategia:	Sólo programación genética
Estructura de datos:	slp
Operador de calificación:	$\{MSE, AIC, BIC, SRM\}$
Operador de cruce:	slp
Operador de mutación:	slp
Varianza del ruido:	1

En la Tabla 9.2 podemos observar como la tasa de error (t_e) y tasa de éxito (t_s) asociada a los métodos de calificación *AIC* y *BIC* son similares con ambas muestras lo cual contradice los resultados obtenidos en [CY03]. Sin embargo las diferencias se encuentran en los métodos basados en *MSE* y *SRM*. En los experimentos basados en muestras polinomiales la tasa de solución es similar (y muy baja) pues es muy complicado

ajustar los coeficientes si además hay ruido. Sin embargo, claramente se observa como el procedimiento de calificación basado en MSE se sobre-ajusta de manera desproporcionada a la muestra y de ahí la tasa de error tan pronunciada.

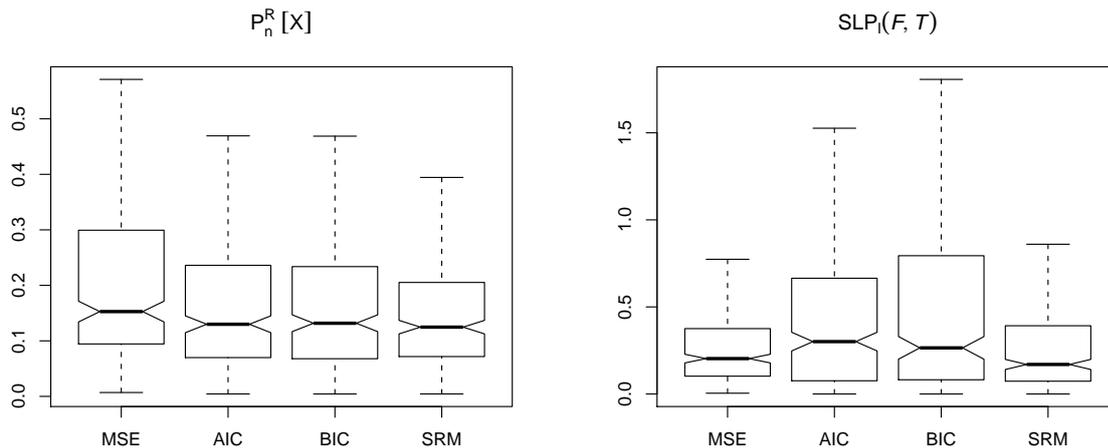
Sin embargo la cosa cambia cuando observamos los resultados de los experimentos basados en muestras provenientes de slp 's aleatorios. En este caso es algo más sencillo encontrar el modelo correcto principalmente debido a que el modelo se encuentra en el espacio de búsqueda y porque hay pocas constantes que ajustar. Esto produce que la calificación basada en MSE no se sobre-ajuste (porque no puede) y presente resultados similares al resto o incluso mejores. De todas formas se puede ver como la calificación basada en SRM produce mejores resultados que el resto.

Cuadro 9.2: Tasas de error y de solución: *métodos de calificación con ruido.*

$P_n^{\mathbb{R}}[X]$	t_e	t_s	$SLP_l(\mathcal{F}, \mathcal{T})$	t_e	t_s
MSE	12 %	0,33 %	MSE	6 %	9,33 %
AIC	4,33 %	0,33 %	AIC	8,67 %	4,33 %
BIC	4 %	0,33 %	BIC	7,33 %	4,33 %
SRM	6,33 %	0,33 %	SRM	5,67 %	8 %

Los diagramas de cajas de la Figura 9.1 y las Tablas 9.3 y 9.4 confirman lo anteriormente expuesto.

Figura 9.1: Diagramas de Cajas y Bigotes: *métodos de calificación con ruido.*



Como en la experimentación anterior, observamos en la columna referida al mínimo en las muestras generadas mediante slp 's que el los métodos de calificación basados en métodos estadísticos así como el basado en SRM han encontrado el modelo *exacto*.

Cuadro 9.3: Resumen estadístico I: *métodos de calificación con ruido.*

$P_n^{\mathbb{R}}[X]$	μ	σ	$SLP_l(\mathcal{F}, \mathcal{T})$	μ	σ
MSE	0,17	0,12	MSE	0,24	0,18
AIC	0,15	0,11	AIC	0,35	0,36
BIC	0,15	0,11	BIC	0,43	0,47
SRM	0,13	$8,62 \cdot 10^{-2}$	SRM	0,22	0,21

Cuadro 9.4: Resumen estadístico II: *métodos de calificación con ruido.*

$P_n^{\mathbb{R}}[X]$	mín	q_1	q_2	q_3	máx
MSE	$6,97 \cdot 10^{-3}$	$9,05 \cdot 10^{-2}$	0,13	0,22	0,57
AIC	$4,39 \cdot 10^{-3}$	$6,65 \cdot 10^{-2}$	0,12	0,22	0,47
BIC	$4,37 \cdot 10^{-3}$	$6,71 \cdot 10^{-2}$	0,13	0,22	0,47
SRM	$4,4 \cdot 10^{-3}$	$6,92 \cdot 10^{-2}$	0,12	0,18	0,39

$SLP_l(\mathcal{F}, \mathcal{T})$	mín	q_1	q_2	q_3	máx
MSE	$4,31 \cdot 10^{-3}$	$9,99 \cdot 10^{-2}$	0,19	0,33	0,77
AIC	0	$7,15 \cdot 10^{-2}$	0,25	0,5	1,53
BIC	0	$7,28 \cdot 10^{-2}$	0,24	0,63	1,81
SRM	0	$7,02 \cdot 10^{-2}$	0,15	0,34	0,86

Para finalizar, los contrastes de hipótesis no dejan lugar a dudas. Para cualquier valor razonable de la significancia α los métodos de calificación basados en métodos estadísticos así como el basado en *SRM* son significativamente mejores que el basado en *MSE*. Además, como comentamos anteriormente, el método de calificación basado en *SRM* es superior a los métodos estadísticos.

Cuadro 9.5: Contraste de Hipótesis: *métodos de calificación con ruido.*

$P_n^{\mathbb{R}}[X]$	MSE	AIC	BIC	SRM
MSE	1	0,94	0,93	1
AIC	$8,81 \cdot 10^{-3}$	1	0,73	0,8
BIC	$5,82 \cdot 10^{-3}$	0,91	1	0,94
SRM	$3,44 \cdot 10^{-3}$	$7,02 \cdot 10^{-2}$	$7,16 \cdot 10^{-2}$	1

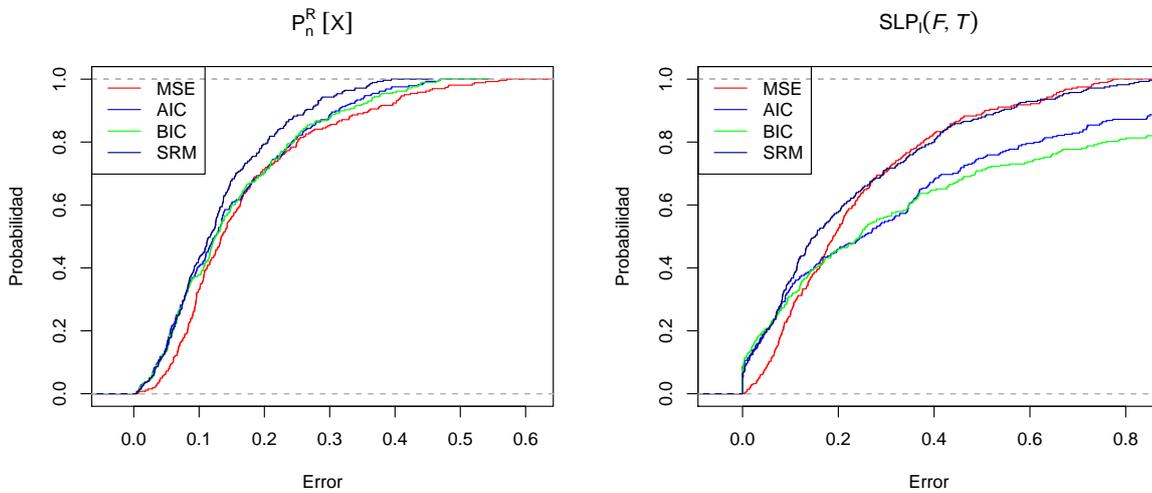
$SLP_l(\mathcal{F}, \mathcal{T})$	MSE	AIC	BIC	SRM
MSE	1	$4,67 \cdot 10^{-5}$	$1,67 \cdot 10^{-5}$	0,81
AIC	$1,51 \cdot 10^{-2}$	1	0,11	0,77
BIC	$5 \cdot 10^{-3}$	0,75	1	0,63
SRM	$8,72 \cdot 10^{-3}$	$2,53 \cdot 10^{-4}$	$2,19 \cdot 10^{-5}$	1

Sin embargo hay un resultado cuanto menos curioso y que requiere una explicación. Observando la tabla con los resultados del contraste de hipótesis en el caso de muestras generadas mediante *slp* aleatorios se observa que, tanto el método de calificación *AIC* como el método *BIC*, tiene resultados significativamente mejores que el método basado

en el MSE . A su vez, el método basado en MSE tiene resultados significativamente mejores que los basados en propiedades estadísticas. ¿Cómo puede ser esto posible?

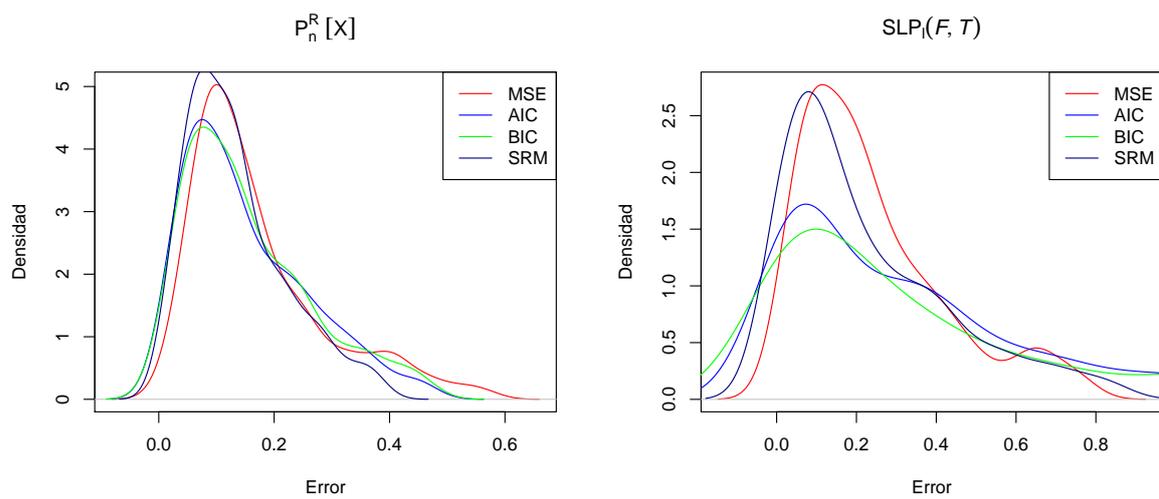
Observando la gráfica de las funciones de distribución empíricas ($ecdf$) (Figura 9.2) se puede ver como la $ecdf$ asociada al método MSE es inferior a los basados en propiedades estadísticas en la región de la izquierda, la cual corresponde con calificaciones próximas a cero. Sin embargo la tendencia cambia hacia la mitad de la gráfica y en la región de la derecha son los métodos de calificación basados en propiedades estadísticas los que tiene peores resultados. Esto significa que dichos métodos presentan mayor probabilidad

Figura 9.2: Función de densidad empírica: *métodos de calificación con ruido.*



de tener mejores resultados mientras que el método de calificación basado en MSE tiene menor probabilidad de tener peores resultados. Esta última afirmación se observa con claridad en la Figura 9.3 donde se representa la función de probabilidad empírica obtenida por todos los métodos. Como, a la vista del problema que estamos estudiando, es preferible tener más resultados en la parte izquierda de la gráfica consideramos justificada nuestra anterior conclusión.

Figura 9.3: Función de densidad empírica: *métodos de calificación con ruido*.



Capítulo 10

Co-evolución

Première Loi.

Dans tout animal qui n'a point dépassé le terme de ses développemens, l'emploi plus fréquent et soutenu d'un organe quelconque, fortifie peu à peu cet organe, le développe, l'agrandit, et lui donne une puissance proportionnée à la durée de cet emploi; tandis que le défaut constant d'usage de tel organe, l'affoiblit insensiblement, le détériore, diminue progressivement ses facultés, et finit par le faire disparaître ¹.

Jean-Baptiste Lamarck [Lam09, Vol. 1, pág 235]

En el anterior capítulo introducimos la noción de VCD y la usamos para modificar el método de calificación con el fin de no sobre-ajustarnos a la muestra. Como se puede observar en las conclusiones del Teorema 9.0.1 el parámetro relevante a la hora de calcular la VCD de los *slp's* resulta ser el número de operaciones no escalares. Esto sugiere que, en algún sentido, las operaciones escalares son *gratis*. Es por lo que tratamos de fomentar el uso de dichas operaciones en nuestro modelo. Para ello aumentamos el conjunto de terminales \mathcal{T} e introducimos en él un tercer conjunto de terminales que denominaremos *paramétricas*. De esta forma el conjunto de terminales quedaría:

$$\mathcal{T} = V \cup C \cup P,$$

donde $V := \{x_1, \dots, x_n\}$ denota el conjunto de variables, $C := \{c_1, \dots, c_l\}$ denota el conjunto de constantes numéricas y finalmente $P := \{p_1, \dots, p_q\}$ denota el conjunto de parámetros. Dichos parámetros son, a todos los efectos, constantes numéricas que toman valores en un cierto conjunto, sobre las que usaremos algún mecanismo para optimizar el desempeño del modelo. Para ver otra visión distinta pero con el mismo espíritu véase [PMCC09].

En este capítulo mostraremos un algoritmo evolutivo para optimizar los valores de dichos parámetros. Recordemos que nuestra máquina de aprendizaje automático estaba

¹Primera Ley:

En cada animal que no ha superado los límites de su desarrollo, el uso frecuente de un órgano lo fortalece, lo desarrolla y lo amplía dándole una potencia proporcional a la cantidad de tiempo que ha sido usado, mientras que el desuso de cualquier órgano lo debilita y lo deteriora progresivamente hasta que pierde su capacidad funcional y desaparece.

basada en un método de programación genética. En nuestro caso, evolucionábamos una población

$$Pop_{\Gamma} = \{\Gamma_1, \dots, \Gamma_N\}$$

constituída por N *slp's* sobre los conjuntos \mathcal{F} y $\mathcal{T} = V \cup C \cup P$.

Paralelamente al proceso evolutivo mediante programación genética de los *slp's* realizaremos un algoritmo evolutivo en el que las constantes serán optimizadas. Para ello contamos también con una población de parámetros

$$Pop_{\mathcal{P}} = \{\mathbf{p}_1, \dots, \mathbf{p}_N\}$$

en la que cada \mathbf{p}_i es un vector de q componentes con $\mathbf{p}_i \in [a, b]^q$ para todo i , $1 \leq i \leq N$.

Además, en la Sección 10.3 presentamos un estudio empírico del desempeño entre varias formas de entrelazar ambos procesos evolutivos.

10.1 Programación Genética con *slp's* paramétricos

Como ya hemos visto anteriormente ahora contamos con una población de parámetros $Pop_{\mathcal{P}}$ a optimizar. Pasamos ahora a enunciar las principales características que presenta el algoritmo evolutivo con el que optimizaremos los parámetros de los *slp's*. Inicialmente generaremos dicha población con elementos distribuidos uniformemente en $[a, b]^q$.

Parece razonable que la definición del operador de calificación de un elemento $\mathbf{p} \in Pop_{\mathcal{P}}$ esté relacionada de alguna forma con la calificación de los *slp's* que están siendo evolucionados. De esta forma, dada una muestra

$$z = (x_i, y_i) \in \mathbb{R}^n \times \mathbb{R}, \quad 1 \leq i \leq m,$$

una primera forma de calificación para un vector de parámetros \mathbf{p} podría ser:

$$\mathcal{F}it_z^{\mathcal{P}}(\mathbf{p}) = \min_{1 \leq i \leq N} \{\mathcal{F}it_z(\Gamma_i^{\mathcal{P}})\},$$

donde $\Gamma_i^{\mathcal{P}}$ es el *slp* Γ_i en el que se han sustituido los parámetros P por sus valores numéricos \mathbf{p} y $\mathcal{F}it_z(\Gamma_i^{\mathcal{P}})$ representa el operador de calificación para *slp's* que fue definido previamente en la Ecuación (8.2). A este método de calificación de constantes lo denominaremos normal.

Rápidamente se observa que para calificar elementos \mathbf{p} es necesario calificar toda la población de *slp's* que está siendo evolucionada. Esto puede ser (y de hecho lo es, como se verá más adelante en los sección de resultados experimentales) un derroche de recursos. Es por ello que introducimos una nueva forma de calificación que en cierto sentido *aproxima* la anterior. De hecho lo que sugerimos es usar solamente un subconjunto de la población Pop_{Γ} a la hora de realizar la calificación de los parámetros. Trabajos anteriores (véase [WLJ01]) sugieren dos métodos para elegir dicho subconjunto:

- Seleccionar el mejor elemento de la actual población de *slp's* (que en lo siguiente denominaremos *élite*). Esto es:

$$\mathcal{F}it_z^e(\mathbf{p}) = \mathcal{F}it_z(\Gamma_{\text{élite}}^{\mathcal{P}}),$$

donde $\Gamma_{\text{élite}}^{\mathbf{p}}$ representa el mejor *slp* de la actual población en el que se han sustituido los parámetros por sus valores \mathbf{p} . A este método de calificación de constantes lo denominaremos élite.

- Seleccionar dos elementos en la actual población de *slp*'s, la élite y otro elemento elegido al azar y usar como calificación la media o el mejor valor entre ellos. Es decir, uno de los siguientes:

$$\begin{aligned}\mathcal{F}it_z^e(\mathbf{p}) &= \text{mín} \{ \mathcal{F}it_z(\Gamma_{\text{élite}}^{\mathbf{p}}), \mathcal{F}it_z(\Gamma_r^{\mathbf{p}}) \}, \\ \mathcal{F}it_z^e(\mathbf{p}) &= \frac{1}{2} (\mathcal{F}it_z(\Gamma_{\text{élite}}^{\mathbf{p}}) + \mathcal{F}it_z(\Gamma_r^{\mathbf{p}})),\end{aligned}$$

donde $\Gamma_{\text{élite}}^{\mathbf{p}}$ representa el mejor *slp* de la actual población y $\Gamma_r^{\mathbf{p}}$ otro elemento distinto en los que se han sustituido los parámetros por sus valores \mathbf{p} . Al primero de los métodos de calificación de constantes enunciado lo denominaremos mínimo mientras que al segundo lo denominaremos media.

En la Sección 10.3 realizaremos un estudio empírico entre los tres procedimientos.

Para el operador de cruce usaremos el llamado cruce aritmético (véase [Sch81]). Este operador de cruce funciona de la siguiente forma: dados $\mathbf{p}_1, \mathbf{p}_2 \in [a, b]^q$, el cruce aritmético produce dos hijos \mathbf{p}'_1 y \mathbf{p}'_2 que son una combinación convexa de sus padres. Es decir:

$$\mathbf{p}'_1 = \lambda \cdot \mathbf{p}_1 + (1 - \lambda) \cdot \mathbf{p}_2 \quad \mathbf{p}'_2 = \lambda \cdot \mathbf{p}_2 + (1 - \lambda) \cdot \mathbf{p}_1,$$

donde λ es un número real escogido de forma aleatoria en el intervalo $[0, 1]$ en cada operación de cruce.

Debido a que nuestro espacio de búsqueda $[a, b]^q$ es convexo, definimos un operador de mutación no uniforme (véase [MLS94]). En nuestro caso, el operador de mutación generará el hijo \mathbf{p}' a partir del padre \mathbf{p} escogiendo uniformemente una de las siguientes posibilidades:

$$\mathbf{p}'_k = \mathbf{p}_k + \Delta(t, b - \mathbf{p}_k), \quad \mathbf{p}'_k = \mathbf{p}_k - \Delta(t, \mathbf{p}_k - a),$$

para $k = 1, \dots, q$, t el número de generación actual y la función Δ definida de la siguiente manera:

$$\Delta(t, y) := yr \left(1 - \frac{t}{T} \right),$$

con r un número elegido al azar en $[0, 1]$ y T representando el máximo número de generaciones.

Nótese que la función $\Delta(t, y)$ devuelve un valor en aleatorio en $[0, y]$ de tal forma que la probabilidad de que $\Delta(t, y)$ está cerca de 0 aumenta con el número de generaciones t . Esta propiedad permite explorar libremente el espacio de búsqueda en las primeras generaciones transformándose, con el paso de las generaciones, en una búsqueda local. En los experimentos realizados por Michalewicz, et al. (véase [MLS94]), fue usada la función:

$$\Delta(t, y) := yr \left(1 - \frac{t}{T} \right)^b,$$

donde b es un parámetro que determina el grado de no uniformidad del operador.

El operador de selección elegido en este caso el método q -torneo mientras que el operador de reemplazo es el mismo definido en la Definición 8.2.7.

10.2 Algoritmos de co-evolución cooperativa

En contraposición con la co-evolución competitiva, en la que los individuos de las distintas poblaciones tienen calificaciones independientes y compiten por los recursos, en la co-evolución cooperativa elementos de ambas poblaciones son necesarios a la hora de dar una calificación. Así, ambas poblaciones evolucionan libremente y llegado el momento de obtener la calificación de un elemento de una de las poblaciones ya hemos visto los mecanismos que usaremos para escoger los colaboradores de la otra población.

En un lenguaje de muy alto nivel, el proceso evolutivo que proponemos y que se ha implementado es el siguiente:

Pseudocódigo 10.1: *Pseudocódigo del proceso co-evolutivo.*

```
PopG := generar población inicial de slp's
PopP := generar población inicial de parámetros
escoger aleatoriamente slpélite
escoger aleatoriamente pélite
calificar población inicial de slp's
mientras (no condición de finalización) hacer
  realizar un turno de GP
  seleccionar slpélite
  realizar un turno de AE
  seleccionar pélite
fin
```

En donde entendemos por *realizar un turno* de *GP* o *AE* la realización de ininterrumpida de una o varias generaciones evolutivas como las descritas en el Pseudo-código 3 bien sobre la población de *slp's* (programación genética) o bien sobre la población de parámetros (algoritmo evolutivo).

Distinguiremos dos procesos co-evolutivos distintos en función de como se alternen los turnos de programación genética y algoritmo evolutivo:

Rondas: Se realizan un número fijo de generaciones *a* de programación genética seguido de otro número fijo (que no tiene que ser idéntico) de generaciones *b* de algoritmo evolutivo hasta finalizar las generaciones. Este método lo denotaremos por *Ga – Eb*.

Separadas: Se separa la ejecución en dos turnos. En un primer turno se realiza toda las generaciones de programación genética asignadas (generalmente un porcentaje *c* del total) y posteriormente se realiza el resto de las generaciones hasta el final de algoritmo evolutivo. Este método lo denotaremos por *S – c*.

En la Sección 10.3 realizaremos un estudio empírico entre ambos procedimientos. En [AMB09] presentamos otros resultados similares.

10.3 Resultados experimentales

En las siguientes páginas realizaremos un estudio empírico sobre los mecanismos de calificación adecuados para el algoritmo evolutivo. Demostraremos que no hay diferencias significativas entre ellos y por lo tanto usaremos el más simple y fácil de calcular (el anteriormente llamado élite). Seguidamente comprobaremos como hay ligeras evidencias de que los procedimientos basados en la evolución de *slp's* más el ajuste de constantes produce resultados significativamente mejores que solamente realizando la parte de programación genética.

Para llevar a cabo dicho estudio seguiremos el protocolo propuesto en el Anexo B. Pasamos ahora a detallar los parámetros particulares de cada uno de los experimentos.

10.3.1 Calificación de constantes

Los parámetros específicos de este experimento son:

Cuadro 10.1: Parámetros específicos: *calificación de constantes*.

Tipo de estrategia: S-90
 Estructura de datos: *slp*
 Operador de calificación: *MSE* y {normal, élite, media, mínimo}
 Operador de cruce: *slp* y aritmético
 Operador de mutación: *slp* y no uniforme

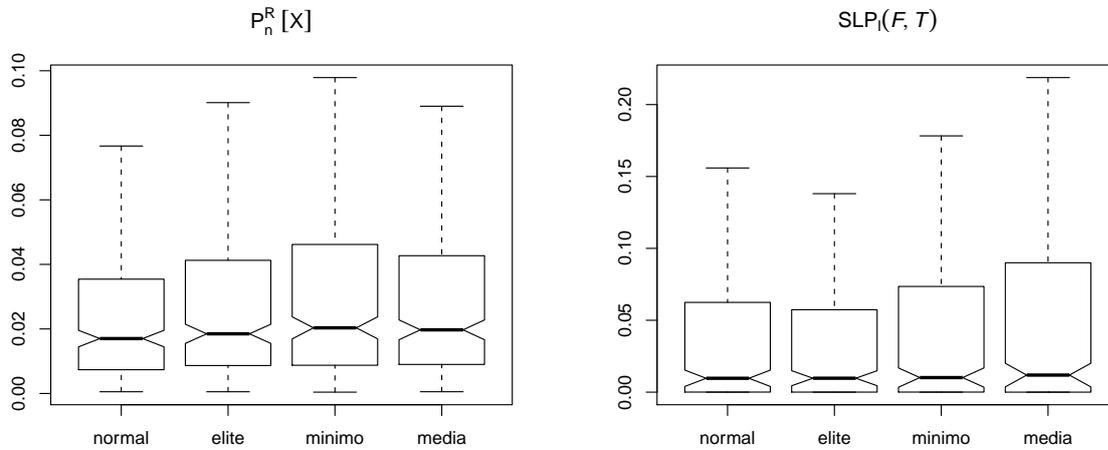
En la Tabla 10.2 volvemos a observar el mismo patrón que observamos en la Sección 8.3: la tasa de error y la tasa de éxito son muy similares entre todos los procesos de calificación y en el caso de las muestras generadas sobre *slp's* aleatorios peores que sobre las muestras generadas mediante polinomios.

Cuadro 10.2: Tasas de error y de solución: *calificación de constantes*.

$P_n^R[X]$	t_e	t_s	$SLP_l(\mathcal{F}, \mathcal{T})$	t_e	t_s
normal	9,67 %	99,67 %	normal	18,67 %	98 %
elite	9 %	100 %	elite	16,67 %	96 %
minimo	7 %	100 %	minimo	16 %	97 %
media	9 %	99,67 %	media	15 %	96 %

Sin embargo en los diagramas de cajas de la Figura 10.1 se aprecia una ligera diferencia a favor de los métodos normal y élite frente a los otros dos. Esto último se observa, sobre todo, en las muestras generadas sobre *slp's* aleatorios.

Figura 10.1: Diagramas de Cajas y Bigotes: *calificación de constantes.*



Las Tablas 10.3 y 10.4 no introducen ninguna diferencia significativa respecto a lo anteriormente mostrado.

Cuadro 10.3: Resumen estadístico I: *calificación de constantes.*

$P_n^R[X]$	μ	σ	$SLP_l(\mathcal{F}, \mathcal{T})$	μ	σ
normal	$1,98 \cdot 10^{-2}$	$1,69 \cdot 10^{-2}$	normal	$1,74 \cdot 10^{-2}$	$2,79 \cdot 10^{-2}$
elite	$2,28 \cdot 10^{-2}$	$2,01 \cdot 10^{-2}$	elite	$1,91 \cdot 10^{-2}$	$2,93 \cdot 10^{-2}$
minimo	$2,67 \cdot 10^{-2}$	$2,36 \cdot 10^{-2}$	minimo	$2,26 \cdot 10^{-2}$	$3,72 \cdot 10^{-2}$
media	$2,41 \cdot 10^{-2}$	$2,09 \cdot 10^{-2}$	media	$2,99 \cdot 10^{-2}$	$4,99 \cdot 10^{-2}$

Cuadro 10.4: Resumen estadístico II: *calificación de constantes.*

$P_n^R[X]$	mín	q_1	q_2	q_3	máx
normal	$5,42 \cdot 10^{-4}$	$6,91 \cdot 10^{-3}$	$1,44 \cdot 10^{-2}$	$2,83 \cdot 10^{-2}$	$7,67 \cdot 10^{-2}$
elite	$5,42 \cdot 10^{-4}$	$8,07 \cdot 10^{-3}$	$1,71 \cdot 10^{-2}$	$3,13 \cdot 10^{-2}$	$8,76 \cdot 10^{-2}$
minimo	$4,13 \cdot 10^{-4}$	$8,4 \cdot 10^{-3}$	$1,85 \cdot 10^{-2}$	$3,93 \cdot 10^{-2}$	$9,79 \cdot 10^{-2}$
media	$5,44 \cdot 10^{-4}$	$8,13 \cdot 10^{-3}$	$1,69 \cdot 10^{-2}$	$3,49 \cdot 10^{-2}$	$8,9 \cdot 10^{-2}$

$SLP_l(\mathcal{F}, \mathcal{T})$	mín	q_1	q_2	q_3	máx
normal	0	$4,58 \cdot 10^{-33}$	$3,99 \cdot 10^{-3}$	$2,52 \cdot 10^{-2}$	0,15
elite	0	$3,55 \cdot 10^{-33}$	$3,93 \cdot 10^{-3}$	$3 \cdot 10^{-2}$	0,14
minimo	0	$4,06 \cdot 10^{-33}$	$4,57 \cdot 10^{-3}$	$2,51 \cdot 10^{-2}$	0,18
media	0	$1,94 \cdot 10^{-27}$	$6,14 \cdot 10^{-3}$	$3,56 \cdot 10^{-2}$	0,22

En esta ocasión los contrastes de hipótesis no nos despejan mucho las incógnitas. Solamente para en algunos casos obtenemos diferencias significativa como para realizar alguna afirmación.

Cuadro 10.5: Contraste de Hipótesis: *calificación de constantes*.

$P_n^R[X]$	normal	elite	minimo	media
normal	1	0,2	$8,72 \cdot 10^{-3}$	0,14
elite	0,97	1	$8,49 \cdot 10^{-2}$	0,49
minimo	0,97	0,97	1	0,87
media	0,99	0,94	0,31	1

$SLP_l(\mathcal{F}, \mathcal{T})$	normal	elite	minimo	media
normal	1	0,63	0,3	$7,54 \cdot 10^{-2}$
elite	0,86	1	0,52	0,17
minimo	0,93	0,85	1	0,27
media	0,75	0,95	0,78	1

En el caso de las muestras generadas por polinomios aleatorios observamos que la calificación normal y elite son superiores a calificación mínimo mientras que en el caso de muestras generadas por *slp*'s aleatorios es la calificación normal la que es superior a la calificación media.

A tenor de estos resultados queda claro que es preferible usar el mecanismos de calificación elite pues a tenor de nuestros datos no es significativamente peor al mecanismo de calificación normal y sin embargo es computacionalmente menos intensivo.

10.3.2 Algoritmos de co-evolución

Los parámetros específicos de este experimento son:

Cuadro 10.6: Parámetros específicos: *algoritmos de co-evolución*.

Tipo de estrategia:	{ <i>GP</i> , S-90, S-75, G20-E5, G30-E}
Estructura de datos:	<i>slp</i>
Operador de calificación:	<i>MSE</i> y elite
Operador de cruce:	<i>slp</i> y aritmético
Operador de mutación:	<i>slp</i> y no uniforme

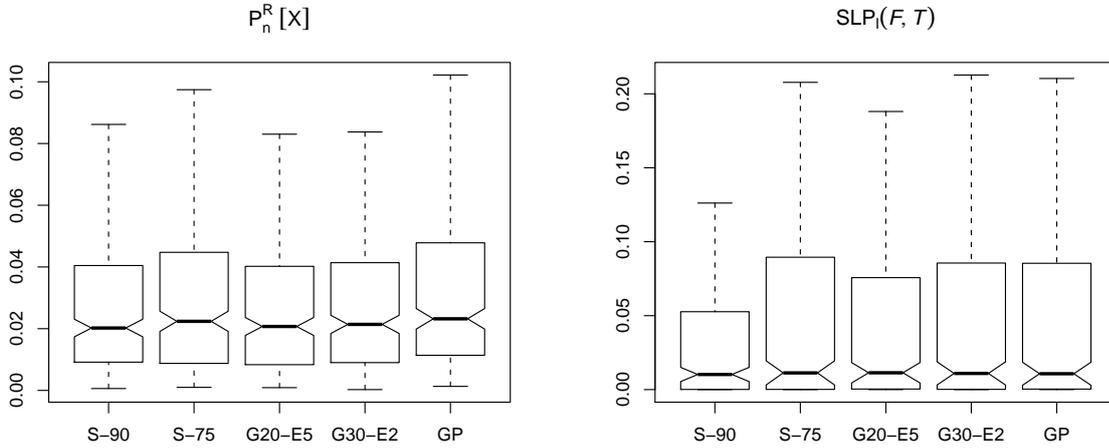
En la Tabla 10.7 podemos observar que tanto la tasa de error como la tasa de éxito son muy similares a las obtenidos en anteriores problemas y no nos dan ninguna información.

Cuadro 10.7: Tasas de error y de solución: *algoritmos de co-evolución*.

$P_n^R[X]$	t_e	t_s	$SLP_l(\mathcal{F}, \mathcal{T})$	t_e	t_s
S-90	6,67 %	100 %	S-90	16,67 %	98,33 %
S-75	5,67 %	100 %	S-75	14,67 %	99 %
G20-E5	8,67 %	100 %	G20-E5	16 %	97,67 %
G30-E2	8,33 %	100 %	G30-E2	16 %	98 %
GP	7,67 %	100 %	GP	14 %	96,67 %

Ni los diagramas de cajas ni las tablas de estadísticos revelan ninguna diferencia en comportamiento en ninguna de las muestras (Figura 10.2 y Tablas 10.8 y 10.9).

Figura 10.2: Diagramas de Cajas y Bigotes: *algoritmos de co-evolución.*



Cuadro 10.8: Resumen estadístico I: *algoritmos de co-evolución.*

$P_n^R[X]$	μ	σ	$SLP_l(\mathcal{F}, \mathcal{T})$	μ	σ
S-90	$2,43 \cdot 10^{-2}$	$2,01 \cdot 10^{-2}$	S-90	$1,79 \cdot 10^{-2}$	$2,69 \cdot 10^{-2}$
S-75	$2,69 \cdot 10^{-2}$	$2,24 \cdot 10^{-2}$	S-75	$2,79 \cdot 10^{-2}$	$4,53 \cdot 10^{-2}$
G20-E5	$2,27 \cdot 10^{-2}$	$1,8 \cdot 10^{-2}$	G20-E5	$2,44 \cdot 10^{-2}$	$3,86 \cdot 10^{-2}$
G30-E2	$2,38 \cdot 10^{-2}$	$2 \cdot 10^{-2}$	G30-E2	$2,52 \cdot 10^{-2}$	$4,27 \cdot 10^{-2}$
GP	$2,83 \cdot 10^{-2}$	$2,34 \cdot 10^{-2}$	GP	$2,75 \cdot 10^{-2}$	$4,35 \cdot 10^{-2}$

Cuadro 10.9: Resumen estadístico II: *algoritmos de co-evolución.*

$P_n^R[X]$	mín	q_1	q_2	q_3	máx
S-90	$5,98 \cdot 10^{-4}$	$8,68 \cdot 10^{-3}$	$1,81 \cdot 10^{-2}$	$3,39 \cdot 10^{-2}$	$8,62 \cdot 10^{-2}$
S-75	$9,92 \cdot 10^{-4}$	$8,38 \cdot 10^{-3}$	$2 \cdot 10^{-2}$	$3,97 \cdot 10^{-2}$	$9,74 \cdot 10^{-2}$
G20-E5	$8,87 \cdot 10^{-4}$	$7,97 \cdot 10^{-3}$	$1,77 \cdot 10^{-2}$	$3,47 \cdot 10^{-2}$	$8,3 \cdot 10^{-2}$
G30-E2	$2,59 \cdot 10^{-4}$	$8,12 \cdot 10^{-3}$	$1,59 \cdot 10^{-2}$	$3,56 \cdot 10^{-2}$	$8,37 \cdot 10^{-2}$
GP	$1,31 \cdot 10^{-3}$	$1,04 \cdot 10^{-2}$	$2,14 \cdot 10^{-2}$	$4,02 \cdot 10^{-2}$	0,1

$SLP_l(\mathcal{F}, \mathcal{T})$	mín	q_1	q_2	q_3	máx
S-90	0	$1,56 \cdot 10^{-21}$	$3,67 \cdot 10^{-3}$	$2,77 \cdot 10^{-2}$	0,13
S-75	0	$2,44 \cdot 10^{-30}$	$5,45 \cdot 10^{-3}$	$3,32 \cdot 10^{-2}$	0,21
G20-E5	0	$3,26 \cdot 10^{-5}$	$6,39 \cdot 10^{-3}$	$3,07 \cdot 10^{-2}$	0,19
G30-E2	0	$4,22 \cdot 10^{-10}$	$5,38 \cdot 10^{-3}$	$2,64 \cdot 10^{-2}$	0,21
GP	0	$4,79 \cdot 10^{-5}$	$6,04 \cdot 10^{-3}$	$3,5 \cdot 10^{-2}$	0,21

Los contraste de hipótesis tampoco arrojan mucha luz. En el caso de polinomios aleatorios si que aparecen varios métodos con un rendimiento superior al clásico de programación genética pero en el caso de muestras generadas mediante *slp's* aleatorios no aparecen diferencias significativas. Esto último es normal pues en el caso de las muestras generadas por polinomios aleatorios hay muchos más constantes a ajustar que en el caso de las muestras generadas mediante *slp's*.

Cuadro 10.10: Contraste de Hipótesis: *algoritmos de co-evolución*.

$P_n^{\mathbb{R}}[X]$	S-90	S-75	G20-E5	G30-E2	GP
S-90	1	0,13	0,89	0,83	0,11
S-75	0,68	1	0,87	0,98	0,3
G20-E5	0,48	$4,88 \cdot 10^{-2}$	1	0,4	$1,86 \cdot 10^{-2}$
G30-E2	0,5	0,16	0,65	1	$2,58 \cdot 10^{-2}$
GP	0,96	0,91	1	1	1

$SLP_l(\mathcal{F}, \mathcal{T})$	S-90	S-75	G20-E5	G30-E2	GP
S-90	1	0,1	0,15	0,24	0,13
S-75	1	1	0,73	0,94	0,65
G20-E5	1	0,59	1	0,78	0,55
G30-E2	0,89	0,45	0,51	1	0,4
GP	1	0,82	0,75	0,98	1

Cabe resaltar que es probable que en estos experimentos las constantes no sean muy relevantes y por lo tanto no haya diferencias significativas. O bien, es posible que el conjunto de parámetros escogido no sea adecuado para estas muestras en particular. En [AMB09, BAdP⁺10] presentamos otro conjunto de resultados en los que obtenemos conclusiones similares excepto en un experimento en el cual, las constantes si que eran muy relevantes. Emplazamos al lector a dicho artículo para conocer los detalles.

Capítulo 11

Trabajos Futuros

I can't be as confident about computer science as I can about biology. Biology easily has 500 years of exciting problems to work on. It's at that level¹.

Donald Knuth [Knu93]

En las anteriores capítulos hemos introducido una nueva estructura de datos, los *slp's* que codifican las instrucciones a evolucionar usadas en una máquina de aprendizaje automático basada en la programación genética para resolver un problema de regresión simbólica. Así mismo hemos desarrollado los mecanismos evolutivos adaptados a esta estructura de datos y hemos comprobado experimentalmente que dicha estructura de datos es superior a la que tradicionalmente usada en programación genética, los árboles.

Posteriormente hemos estudiado la VCD de los *slp's* y a partir de ella hemos desarrollado un operador de calificación que experimentalmente se muestra muy superior a los operadores de calificación *AIC* y *BIC* que son los usados generalmente cuando la muestra de aprendizaje contiene ruido.

Finalmente hemos ampliado la estructura de datos *slp* para introducir parámetros numéricos que actúan de constantes. A continuación hemos desarrollado varios mecanismos de optimización de dichos parámetros numéricos y hemos realizado un estudio experimental.

Sin embargo queda bastante trabajo por hacer. Uno de las principales problemas que padecen los algoritmo evolutivo y la programación genética es la gran cantidad de parámetros relacionados con el propio esquema algorítmico que deben ser ajustados «a mano». En este sentido creemos que al menos los siguientes parámetros:

- la probabilidad de cruce,
- la probabilidad de mutación y
- el tamaño de la población,

¹No puedo tener tanta confianza en las ciencias de la computación como en la biología. La biología tiene fácilmente 500 años de problemas excitantes en los que trabajar. Es a ese nivel.

podrían ajustarse automáticamente a partir de unos ciertos parámetros iniciales *suficientemente buenos*. En este espíritu creemos que los trabajos de Ting Hu y Wolfgang Banzhaf (véase, por ejemplo, [HB09]) podrían servir de base para realizar un estudio empírico sobre este tema.

En este sentido también nos gustaría eliminar la necesidad de fijar *a priori* la longitud de los *slp*'s a evolucionar. Para ello creemos que es necesario introducir un operador de cruce que permita el cambio de longitud así como un mecanismo de calificación que penalice el código no efectivo (aparte de las operaciones no escalares como vimos en la Sección 9.3). En este sentido ya hemos creado dicho operador de cruce (que hemos denominado *slp* extendido) y que opera de forma similar al *slp* salvo que en este caso sustituimos sub-expresiones en los padres aunque no tengan la misma longitud.

Definición 11.0.1 (Cruce *slp* extendido). Sea $\Gamma_1 = \{u_1, \dots, u_L\}$ y $\Gamma_2 = \{v_1, \dots, v_M\}$ dos *slp*'s sobre los conjuntos \mathcal{F} y \mathcal{T} . Sean k y l dos número natural escogidos al azar tales que $1 \leq k \leq L$ y $1 \leq l \leq M$. Tal y como realizamos en la Definición 8.1.11 consideramos los conjuntos

$$\begin{aligned} S_{u_k} &= \{u_j \in \Gamma_1 : u_j \prec u_k\} = \{u_{j_1}, \dots, u_{j_m}\}, \\ S_{v_l} &= \{v_h \in \Gamma_2 : v_h \prec v_l\} = \{v_{h_1}, \dots, v_{h_n}\}, \end{aligned}$$

con $j_1 < \dots < j_m$ y $h_1 < \dots < h_n$. Tal y como hemos mencionado anteriormente, el conjunto S_{u_k} es el código efectivo de Γ_1 necesario para evaluar la variable no terminal u_k así como S_{v_l} es el código efectivo de Γ_2 necesario para evaluar la variable no terminal v_l .

Sea $K = n - m = |S_{v_l}| - |S_{u_k}|$. A continuación sustituimos las variables no terminales del conjunto S_{u_k} en Γ_1 por las variables no terminales del conjunto S_{v_l} renombrando los índices mediante la siguiente aplicación:

$$\begin{aligned} R : \{u_1, \dots, u_L\} \setminus S_{u_k} \cup S_{v_l} &\longrightarrow \{u'_1, \dots, u'_{L+K}\} \cup \mathcal{T} \\ u_i &\longmapsto \begin{cases} u'_i & \text{si } i < k \text{ y } u_i \notin S_{u_k} \\ u'_{i+K} & \text{si } i > k \text{ y } u_i \notin S_{u_k} \\ w & \text{en otro caso} \end{cases} \\ v_{h_i} &\longmapsto u'_{k+i} \end{aligned}$$

donde w denota una variable no terminal escogida al azar entre los terminales y las variables no terminales con índice menor que i .

Mediante este proceso se obtiene el primer hijo producido a partir de los *slp*'s Γ_1 y Γ_2 . Para producir el segundo hijo se aplica el mismo procedimiento simétricamente.

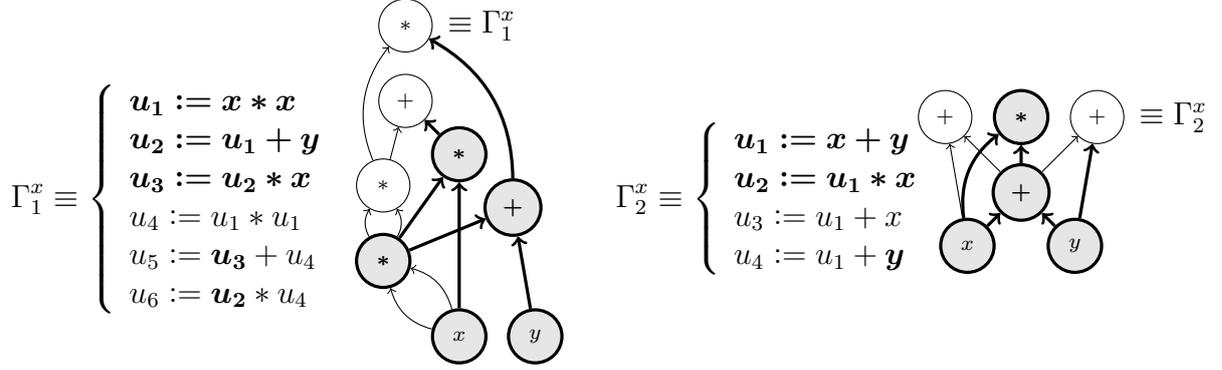
Ejemplo 11.0.2. Consideremos los *slp*'s Γ_1 y Γ_2 procedentes del Ejemplo 8.2.2. Si $k = 3$ y $l = 4$ entonces tenemos que $S_{u_3} = \{u_1, u_3\}$ y $S_{v_4} = \{v_1, v_2, v_4\}$.

Nótese que ya que las variables no terminales u_4 y u_5 de Γ_1 contienen como uno de sus parámetros la variable no terminal u_3 que está en S_{u_3} por lo que será necesario crear dos parámetros de forma aleatoria para generar Γ_1^x . En este caso supongamos que han sido seleccionados u_3 y u_2 según la nueva numeración. Lo mismo sucede para la variable

no terminal v_5 de Γ_2 que tiene como parámetro la variable no terminal v_2 que pertenece a S_{v_4} . En este caso supongamos que hemos escogido el terminal y .

Por lo tanto los slp 's hijos Γ_1^x y Γ_2^x producido por el cruce slp extendido se pueden observar en la Figura 11.1.

Figura 11.1: Slp 's hijos mediante el cruce slp extendido.



Sin embargo no tenemos tan claro como debe ser la penalización que debemos introducir en el operador de calificación. Una penalización muy fuerte sobre el tamaño hace irrelevante el valor del error empírico y produce slp 's con una sola variable no terminal mientras que una muy débil produce slp 's de longitud desproporcionada aunque su código efectivo sea de tamaño «razonable» para el problema a resolver.

Otro campo que parece razonable ampliar es el del ajuste de constantes. La idea es generalizar de nuevo el modelo a fin de introducir operaciones escalares en cada variable no terminal pues recordemos que éstas no afectaban a la capacidad de clasificación de un slp . El problema de dicho enfoque radica en que el número de parámetros sería excesivo para optimizar mediante un enfoque evolutivo como el propuesto en el Capítulo 10. No está claro que un enfoque de optimización basado en técnicas de *descenso de gradiente* sea capaz tampoco de resolverlo pero creemos que sería interesante de abordarlo.

Un primer paso sería realizar un estudio comparativo entre el modelo actual de ajuste de constantes mediante el algoritmo evolutivo y otro nuevo basado en técnicas de descenso de gradiente. Posteriormente pasaríamos a ampliar el modelo el esquema de evaluación de los slp 's eliminando las operaciones $\{+, -\}$ del conjunto de funciones y pasando a estar en cada variable no terminal. De esta forma un slp sobre los conjuntos $\mathcal{F} := \{f_1, \dots, f_m\}$ con aridades $\{a_1, \dots, a_m\}$ y el conjunto de terminales $\mathcal{T} := \{x_1, \dots, x_n\}$ tendría, en su representación como lista de operaciones, con la siguiente estructura:

$$\Gamma \equiv \begin{cases} \mathbf{u}_1 := f_1 \left(\sum_{j=1}^n {}_1c_j^1 x_j + {}_1b_j^1, \dots, \sum_{j=1}^n {}_1c_j^{a_1} x_j + {}_1b_j^{a_1} \right) \\ \vdots \\ \mathbf{u}_k := f_k \left(\sum_{j=1}^n {}_k c_j^1 x_j + {}_k b_j^1, \dots, \sum_{j=1}^n {}_k c_j^{a_k} x_j + {}_k b_j^{a_k} \right) \end{cases}$$

donde ${}_i c_j^l, {}_i b_j^l, 1 \leq i \leq k, 1 \leq j \leq n$ y $1 \leq l \leq a_i$ son constantes numéricas.

Otro punto relacionado con este tema es desarrollar estrategias de *cooperación competitiva* entre poblaciones. Para ello sugerimos la ejecución en paralelo de distintas instancias con poblaciones distintas pero que comparte la misma cota de recursos a usar.

La idea es desarrollar un operador de migración que controle varios de los siguientes aspectos:

- El tamaño de las poblaciones.
- Qué individuos mueren y cuales sobreviven en cada población.
- Qué individuos *emigran* a otras poblaciones.
- Qué poblaciones están comunicadas entre sí.
- Cuantos recursos puede usar cada población.

Como se puede observar, este tema está muy relacionado con el anteriormente descrito acerca de la automatización del ajuste de ciertos parámetros. Consideramos que una «hoja de ruta» adecuada podría ser:

1. Implementar con éxito las ideas de Ting Hu y Wolfgang Banzhaf en nuestro modelo.
2. Desarrollar una implementación *cliente-servidor* que nos permita dividir la población. Este paso sería esencialmente equivalente a lo que ya tenemos pero simplemente tendríamos la oportunidad de ejecutar nuestra implementación en ordenadores de tipo *cluster*.
3. Diseñar e implementar un operador de migración basado en conexiones fijas entre poblaciones, es decir fijando la topología.
4. Diseñar e implementar un operador de migración que evolucione la topología de las conexiones entre las poblaciones reforzando las más prometedoras y eliminando las menos.

Finalmente consideramos interesante probar si esta forma de realizar programación genética se adapta bien a otros problemas como:

- Problemas de clasificación.
- Problemas de predicción de sucesiones.
- Problemas de clústering.

Además sería interesante usar estas técnicas a problemas de escala «real» y no sólo a problemas «académicos».

Capítulo 12

Un algoritmo bioinspirado para el problema XVII de Smale

*Nothing in Biology Makes Sense Except in the Light of Evolution*¹.
Теодосій Григорович Добжанський² [Dob73]

Presentamos ahora varias alternativas con el fin de dar una solución al problema XVII de Smale en el caso real. Todas ellas presenta una complejidad exponencial en el caso peor y desconocida en el caso medio. Cuatro de ellas se basan en distintos algoritmo evolutivo mientras la otra usa una estrategia de «búsqueda exhaustiva». Para justificar su planteamiento necesitaremos demostrar el Teorema 0.0.47. Pero antes comenzaremos discutiendo otras aproximaciones algorítmicas.

Una primera idea podría consistir en un algoritmos evolutivos que tratara de minimizar el valor del sistema de ecuaciones visto como una aplicación, tal como vimos en la introducción. Para ello necesitamos introducir ciertos mecanismos con el fin de construir un algoritmo evolutivo. Básicamente: una forma de representar individuos, un mecanismo de evolución, uno de calificación y una condición de parada.

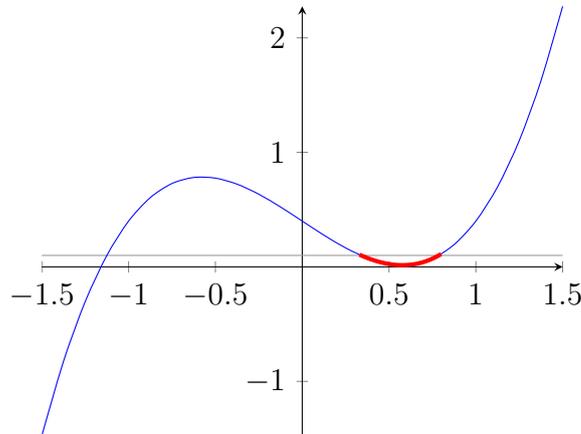
Para la población se podría usar simples vectores que codificarán los puntos candidatos a solución. Como mecanismo de calificación se puede usar simplemente la norma de la evaluación del sistema en el individuo. La condición de parada en este contexto sería encontrar un punto cuya evaluación sea inferior a cierto valor ε o alguna condición que acote el tiempo de ejecución (ya sea en número de generaciones o en polinomios evaluados). Nótese que esto no garantiza que un individuo que cumpla dicha condición sea una raíz. De todas formas mediante este enfoque no es posible hacerlo mejor, téngase en cuenta que la condición $f(x) = 0$ es muy inestable desde un punto de vista numérico (véase por ejemplo [DDH06]).

Para los mecanismos de evolución podría optarse por los vistos anteriormente en el Capítulo 10. Sin embargo el operador de mutación podría ser sustituido por la aplicación del operador de Newton sucesivas veces.

¹Nada tiene sentido en biología excepto a la luz de la evolución.

²Theodosius Dobzhansky.

Figura 12.1: *Ejemplo de sistema con mal comportamiento.*



El principal problema de esta aproximación se encuentra en que no se puede garantizar siquiera que las soluciones obtenidas mediante el algoritmo estén próximas a una raíz. A modo de ejemplo, sistemas como

$$f(x) := x(x - 1)(x + 1) + 0.375 + \varepsilon,$$

podrían presentar falsos positivos cerca del punto 0.5. Además la concentración de individuos de la población alrededor de dicho va a ser elevada al contener un entorno muy grande donde la función de calificación es mínima (véase la Figura 12.1, zona roja). En cierto sentido estamos ante un punto «estable» (o de atracción) en el que NO se encuentra una solución. Esta es una muestra clara de los problemas que puede acarrear la convergencia prematura, pues dicha zona es un mínimo local que no lleva a ninguna solución.

12.1 α^* de un sistema de ecuaciones

Una alternativa para evitar dicho problema es buscar *ceros aproximados* (véase Definición 0.0.7) usando el Teorema 0.0.8. Para ello tomamos $(d) := (d_1, \dots, d_m) \in \mathbb{N}^m$ una lista de grados y f como un sistema de ecuaciones perteneciente al espacio $\mathcal{P}_{(d)}^{\mathbb{R}}$. Sea $x_0 \in \mathbb{R}^n$, el test que se extrae del Teorema 0.0.8 implica calcular el valor $\alpha(f, x_0)$ y chequear la condición

$$\alpha(f, x_0) < \alpha_0$$

donde α_0 toma el valor $\frac{13-3\sqrt{17}}{4}$.

Sin embargo, nos encontramos con la dificultad intrínseca de calcular dicho valor para cada punto que queramos comprobar. Recordemos que para realizar dicho cálculo es necesario evaluar:

- $\beta(f, x_0) = \|D_{x_0}(f)^{-1}f(x_0)\|.$
- $\gamma(f, x_0) = \sup_{k>1} \left\| \frac{D_{x_0}(f)^{-1}D_{x_0}^k(f)}{k!} \right\|^{\frac{1}{k-1}},$

El cálculo de $\beta(f, x_0)$ es directo, pues solo es necesario calcular la matriz Jacobiana del sistema y la norma de un vector. Sin embargo el cálculo de $\gamma(f, x_0)$ es computacionalmente costoso. Para empezar, es necesario calcular las matrices de D tensores, donde D es el máximo de los grados del sistema. Luego debemos evaluar su norma. Solamente la escritura de dichas matrices conlleva $\mathcal{O}((nm)^D)$ operaciones y recordemos que luego es necesario calcular su norma.

Para solventar este problema proponemos calcular el siguiente cantidad:

Definición 12.1.1. Usando la notación anterior. Definimos $\alpha^*(f, x)$ como el siguiente producto:

$$\alpha^*(f, x) := \beta(f, x) \mu_{norm}(f, x) = \frac{D^{3/2}}{2} \|D_x(f)^{-1} f(x)\| \| \|x\| D_x(f)^{-1} \Delta((d)^{1/2}) \| ,$$

donde $\beta(f, x)$ es la cantidad definida anteriormente y $\mu_{norm}(f, x)$ representa el condicionamiento normalizado descrito en la introducción.

Para evaluar esta magnitud solo son necesarios procedimientos básicos de álgebra lineal para matrices y vectores de dimensiones bajas por lo que su evaluación es inmediata. Su principal propiedad ha quedado recogida en el Teorema 0.0.47 enunciado en la introducción. Pasamos pues directamente a demostrar dicho teorema:

12.1.1 Demostración del Teorema 0.0.47

A partir de la definición de cero aproximado (Definición 0.0.7) es evidente que basta con comprobar:

$$\alpha(f, x_0) \leq \alpha_0.$$

Supongamos que se cumple la primera condición. Es decir se cumple que:

$$\alpha^*(f, x_0) \leq \alpha_0.$$

Entonces, usando las cotas del μ -Teorema (Teorema 0.0.13) se tiene automáticamente que $\alpha(f, x_0) \leq \alpha_0$ y por lo tanto x_0 es un cero aproximado.

Supongamos ahora que nos encontramos en el caso

$$\alpha^*(f, x_0) \geq \alpha_0 \text{ pero } \alpha(f, x_0) \leq \alpha_0.$$

Se tiene entonces que x_0 es en realidad un cero aproximado y por lo tanto la iteración de Newton $N_f^k(x_0)$ (véase la Definición 0.0.4) converge cuadráticamente. Es decir,

$$\alpha^*(f, x_k) \leq \frac{D^{3/2}}{2} \frac{1}{2^{2^k-1}} \|x_0 - \zeta\| \mu_{norm}(f, x_0).$$

Sea ahora $0 < \varepsilon \leq \frac{1}{2}$, usando las cotas provenientes de los Corolarios 0.0.17 y 0.0.34 tenemos que:

- $\mu_{norm}(f, x_0)$ esta acotado, con probabilidad $1 - \varepsilon$, por:

$$\mu_{norm}(f, x_0) \leq C(n+1)^{3/2} \left(\frac{N\mathcal{D}}{4\pi} \right)^{1/4},$$

donde C es la constante del Corolario 0.0.17.

- $\|x_0 - \zeta\|$ esta acotado, con probabilidad $1 - \varepsilon$, por:

$$\|x_0 - \zeta\| \leq \frac{\Gamma\left(\frac{1}{4}\right)^2 \mathcal{D}}{\varepsilon^2},$$

donde ζ denota la raíz a la que converge la iteración de Newton y Γ es la función hipergeométrica de mismo nombre.

Tras realizar k iteraciones del operador de Newton con

$$k \geq \log \log \left(L(D(n+1))^{3/2} N^{1/4} \mathcal{D}^{5/4} \varepsilon^{-5/2} \right),$$

donde D denota el máximo de la lista de grados (d), N denota la dimensión como espacio vectorial de $\mathcal{P}_{(d)}^{\mathbb{R}}$, \mathcal{D} denota el número de Bézout y L es la constante

$$L := \frac{\alpha_0 \Gamma\left(\frac{1}{4}\right)^2 C}{(4\pi)^{1/4}},$$

se tiene el resultado esperado. □

Nota 12.1.2. *Nótese que este teorema nos permite realizar técnicas de predicción-corrección que tan buenos resultados han dado en problemas de resolución de sistemas de ecuaciones diferenciales (véase [PFTV92]) y en problemas de optimización lineal (véase [Meh92]). En la Sección 12.4 veremos que dicha técnica también produce muy buenos resultados para la búsqueda de ceros aproximados.*

En este sentido, usamos como predicción del valor de $\alpha(f, x)$ el valor de $\alpha^(f, x)$. En el caso que esta aproximación no sea satisfactoria (i.e. $\alpha^*(f, x) < \alpha_0$) la corregimos aplicando sucesivas veces el operador de Newton.*

12.2 Algoritmos de búsqueda exhaustiva para la resolución del problema XVII de Smale

A partir del Corolario 0.0.17 se tiene que, con probabilidad $1 - \varepsilon$, todas las raíces de un sistema se encuentra en un hiper cubo de tamaño

$$\frac{\Gamma\left(\frac{1}{4}\right)^2 \mathcal{D}}{\varepsilon^2}.$$

Por lo tanto, usando el teorema anterior obtenemos un «test» probabilista que realiza a lo sumo

$$\mathcal{O} \left(\log \log \left(\frac{(Dn)^{3/2} N^{1/4} \mathcal{D}^{5/4}}{\varepsilon^{5/2}} \right) \right)$$

operaciones para comprobar si el punto x_0 de \mathbb{R}^n es un cero aproximado de un sistema f .

Teorema 12.2.1. *Usando la notación anterior. Sea \mathcal{G} el siguiente retículo*

$$\mathcal{G} := \mathbb{Z} \left[\frac{1}{2\sqrt{n}R} \right] \cap H \left[\frac{\Gamma(\frac{1}{4})^2 \mathcal{D}}{\varepsilon^2} \right]$$

donde R es de orden $\mathcal{O}((D(n+1))^{3/2}(N\mathcal{D})^{1/4})$ y $H[a]$ denota el hipercubo de longitud a . Entonces \mathcal{G} tiene, con probabilidad $1 - \varepsilon$, un cero aproximado para cada una de las raíces reales de f .

Demostración. Trivialmente se tiene que la distancia entre dos raíces está acotada por el radio de convergencia dado en el Corolario 0.0.18. Es decir, la distancia media R' entre raíces es

$$R' \geq \frac{K}{(D(n+1))^{3/2}(N\mathcal{D})^{1/4}},$$

donde

$$K := \frac{2u_0\pi^{1/4}}{(6u_0\pi^{1/4} + 2\sqrt{2})C},$$

con u_0 es la constante del Teorema 4.2.1 y C la constante del Teorema 0.0.16. Luego es necesario un retículo con distancia entre elementos inferior a $\frac{1}{2\sqrt{n}R}$, con $R = \frac{1}{R'}$ si queremos asegurarnos que algún punto del retículo se encuentra dentro de cada uno de las regiones de convergencia de cada raíz. \square

Inmediatamente obtenemos el siguiente algoritmo:

Algoritmo 12.2.2 (Búsqueda exhaustiva).

Input: Un sistema de ecuaciones f perteneciente a $\mathcal{P}_{(d)}^{\mathbb{R}}$.

Output: Una lista de ceros aproximados.

1. Para cada elemento x_0 del retículo \mathcal{G} hacer:

- Si $(\alpha^*(f, x_0) < \alpha_0)$ o $(\alpha^*(f, x_k) < \alpha_0)$ guardar x_0 en Z .

2. Devolver Z .

El número de elementos del retículo es de

$$\mathcal{O} \left(\frac{\mathcal{D}^{5/4} N^{1/4} D^{3/2} n^2}{\varepsilon^2} \right).$$

y por lo tanto el número de operaciones está acotado por

$$\mathcal{O} \left(\frac{\mathcal{D}^{5/4} N^{1/4} D^{3/2} n^2}{\varepsilon^2} \log \log \left(\frac{(Dn)^{3/2} N^{1/4} \mathcal{D}^{5/4}}{\varepsilon^{5/2}} \right) \right).$$

Como era de esperar, sufrimos la llamada *maldición de la dimensión* (véase [Pow07]) por lo que el tiempo de ejecución del algoritmo es esencialmente exponencial. Resultados similares han sido obtenidos por Cucker *et ál.* en la serie de artículos [CKMW08, CKMW09].

Para conseguir un algoritmo aplicable en la práctica siguiendo el esquema «exhaustivo» no es buena idea probar sistemáticamente todos los puntos del retículo \mathcal{G} de forma

ordenada pues las cotas presentadas son inferiores (y además muy burdas) con lo cual es altamente probable que si en una región un punto no es un cero aproximado ninguno de sus *vecinos* lo será. Por ello es preferible realizar una búsqueda *adaptativa* refinando el retículo en cada iteración. La literatura sobre métodos adaptativos es extensa pues es usada sistemáticamente en todos los algoritmos de resolución numérica de ecuaciones diferenciales e integrales. Véase [FMM77] o [PLW05], por ejemplo, para una introducción a los métodos adaptativos.

Usando este tipo de técnicas exploramos todo el espacio de búsqueda sin centrarnos demasiado en ninguna parte en concreto y vamos refinando el retículo si a un determinado «zoom» no encontramos los ceros aproximados.

El pseudocódigo de dicho algoritmo sería:

Algoritmo 12.2.3 (Búsqueda exhaustiva con refinamiento del retículo).

Input: *Un sistema de ecuaciones f perteneciente a $\mathcal{P}_{(d)}^{\mathbb{R}}$.*

Output: *Una lista de ceros aproximados.*

1. Mientras $\frac{1}{2^n}$ sea mayor que $\frac{1}{2\sqrt{n}R}$ donde R es la constante definida en el Teorema 12.2.1, definimos \mathcal{G}_0 como el retículo

$$\mathcal{G}_0 := \mathbb{Z} \left[\frac{1}{2^n} \right] \cap_H \left[\frac{\Gamma(\frac{1}{4})^2 \mathcal{D}}{\varepsilon^2} \right]$$

2. Para cada elemento x_0 del retículo \mathcal{G}_0 hacer:

- Si $(\alpha^*(f, x_0) < \alpha_0)$ o $(\alpha^*(f, x_k) < \alpha_0)$ guardar x_0 en Z .

3. Si $\#Z \neq 0$ devolver Z en otro caso hacer $n = n + 1$ y volver al paso 1.

Pero aún con estas mejoras, en la práctica, el tiempo de ejecución sigue siendo demasiado elevado. Es necesario disminuir *aún más* el número de puntos a testar. Es aquí donde entran de nuevo los algoritmos evolutivos.

12.3 Algoritmos evolutivos para la resolución del problema XVII de Smale

Presentamos ahora cuatro alternativas evolutivas basadas en las técnicas vistas en anteriores capítulos de la memoria. Consideramos evolucionar una población de puntos en el retículo para tratar de minimizar la función $\alpha^*(f, x_0)$. Sin embargo, es mucho más fácil usar simplemente puntos en el hipercubo de diámetro $\Gamma(\frac{1}{4})^2 \mathcal{D} \varepsilon^{-2}$ donde, en virtud del Corolario 0.0.17, sabemos que con alta probabilidad se encuentran las raíces del sistema.

Para la primera alternativa usaremos un algoritmo evolutivo genético basado en el Pseudocódigo 3, por lo que debemos definir los siguientes mecanismos para este problema en concreto: una forma de representar individuos, un mecanismo de evolución, uno de calificación y una condición de parada.

Para codificar los elementos de la población usaremos simplemente vectores de números reales. Nótese que dichos puntos ahora no codifican candidatos a raíz sino a cero aproximado.

Por otro lado, los mecanismos de evolución serán:

- Como operador de cruce usaremos el cruce aritmético anteriormente descrito.
- Como operador de mutación usaremos la mutación no uniforme también descrita anteriormente.

Como mecanismo de calificación presentamos dos opciones:

- Usar como calificación el valor de $\alpha^*(f, x)$ a partir de su definición. A esta estrategia de calificación la denominaremos α^* .
- Usar como calificación el valor de $\alpha^*(f, x_k)$ donde x_k denota aplicar k iteraciones del método de Newton sobre el punto x y k es la constante definida en el Teorema 0.0.47. A esta estrategia de calificación la denominaremos α^* corregida (véase la Nota 12.1.2) que usualmente denotaremos por α_c^* .

En función del objetivo deseado podemos guardar todos los individuos cuya calificación sea inferior a α_0 (en este sentido el algoritmo evolutivo será elitista) o simplemente usaremos como condición haber encontrado un cero aproximado. En nuestras pruebas usaremos esta última.

Nota 12.3.1. *La principal ventaja de este algoritmo respecto al anterior algoritmo evolutivo se encuentra en que este último si que garantiza la correctitud de las soluciones. Sin embargo, tampoco podemos dar una cota sobre el número de operaciones, salvo la trivial que se deriva del Algoritmo 12.2.2.*

Actualmente ha surgido una variante de los *algoritmos de escalada* llamada evolución diferencial que presenta resultados sorprendentes para la resolución de problemas de optimización. Véase por ejemplo [Cha08] y [PSL05] para una introducción a los algoritmos basados en evolución diferencial y a los problemas en los que mejores resultados consiguen. Nosotros consideraremos una adaptación/hibridación del Pseudocódigo 2 con el Pseudocódigo 3.

La principal diferencia de este algoritmo respecto al algoritmo evolutivo genético presentado anteriormente se encuentra en el mecanismo de evolución usado. Mientras que en el anterior ejemplo usábamos el cruce aritmético en este caso usaremos un el siguiente cruce (que denominaremos cruce diferencial): Dados $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3 \in [a, b]^q$, el cruce diferencial produce un hijo \mathbf{h} que es una combinación lineal de sus padres. Es decir:

$$\mathbf{h} = \mathbf{p}_1 + \lambda(\mathbf{p}_2 - \mathbf{p}_3),$$

donde λ es un número real escogido de forma aleatoria en el intervalo $[0, 1]$ en cada operación de cruce.

Esta pequeña modificación veremos en la Sección 12.4 que provoca un comportamiento completamente distinto al de los algoritmos evolutivos genéticos.

Por otro lado, otra técnica que también ha presentado muy buenos resultados en problemas de optimización es la conocida como enfriamiento simulado (véase [LA87] para una introducción teórica y práctica a este tipo de algoritmos). Igualmente, como en el caso anterior consideraremos una adaptación/hibridación del Pseudocódigo 2 con el Pseudocódigo 3.

En este caso, la principal diferencia es el método de reemplazo usado. Hemos denominado a dicho método de reemplazo enfriamiento simulado. En esencia, este método permite que los hijos reemplacen a los padres en la siguiente generación siempre que sean mejores pero, en el caso que sean peores, lo harán con una cierta probabilidad que denominaremos *probabilidad de reemplazo*.

Al igual que en el caso anterior, esta pequeña modificación hace que los resultados experimentales cambien muy significativamente, como veremos en la Sección 12.4.

Para terminar, otra técnica que está dando muy buenos resultados es la llamada enjambre de partículas (véase [Pol08] para una revisión sobre el estado del arte). Igualmente, como en los casos anteriores, consideraremos una adaptación/hibridación del Pseudocódigo 4 con el Pseudocódigo 3.

En esta ocasión, la principal diferencia estriba en el método de cruce usado, que hemos denominado cruce enjambre. Este método opera de la siguiente manera: Para cada partícula \mathbf{p}_i del enjambre denotaremos por \mathbf{p}_i^e al vector con la mejor calificación que ha visitado dicha partícula. Además, denotaremos por \mathbf{p}^e al mejor vector que ha conocido el enjambre en su totalidad. Entonces el resultado del cruce enjambre es:

$$\mathbf{h} = \lambda_1 \mathbf{p}_i + \lambda_2 (\mathbf{p}_i^e - \mathbf{p}_i) + \lambda_3 (\mathbf{p}^e - \mathbf{p}_i)$$

donde λ_1 , λ_2 y λ_3 son números reales escogidos de forma aleatoria en el intervalo $[0, 1]$ en cada operación de cruce.

Además, el otro cambio significativo tiene que ver con el mantenimiento de los elementos \mathbf{p}^e y \mathbf{p}_i^e . En este sentido, esta estrategia obligatoriamente es elitista, no solo para guardar la mejor partícula del enjambre \mathbf{p}^e , sino para conservar las mejores posiciones por las que ha pasado una partícula.

Nota 12.3.2. *Nótese que en este contexto cada partícula no es mas que un individuo de la población que a su vez es llamada enjambre.*

En la Sección 12.4 presentaremos los resultados experimentales en los que comparamos las cuatro estrategias con los diferentes métodos de calificación que hemos enunciado a lo largo del capítulo.

12.4 Resultados experimentales

En esta sección presentaremos los detalles de los resultados obtenidos empíricamente al comparar los distintos tipos de algoritmos evolutivos propuestos en la sección anterior. Puesto que en esta ocasión no estamos resolviendo un problema de regresión simbólica sino uno de resolución numérica de ecuaciones polinomiales multivariadas, no tiene sentido el uso del protocolo de experimentación descrito en el Anexo B como en anteriores ocasiones. Como sustituto proponemos el siguiente:

12.4.1 Protocolo de experimentación

Con el fin de obtener resultados estadísticamente significativos generaremos 300 sistemas f de ecuaciones polinomiales con coeficientes reales dados por slp 's sobre el conjunto de funciones $\mathcal{F} = \{+, -, *\}$ y de terminales $\mathcal{T} = \{x_1, \dots, x_n, c_1, c_2\}$ donde x_1, \dots, x_n denotan las variables y c_1, c_2 denotan dos parámetros reales generados siguiendo una distribución uniforme en el intervalo $[-1, 1]$ cuyo valor fijaremos al generar el slp . Estos parámetros actuarán a modo de *coeficientes*. Además, la longitud de los slp 's estará acotada por $5n$, donde n es el número de variables.

Nos aseguraremos que el grado del polinomio resultante sea al menos 2 y que no sea un sistema degenerado. Para ello usaremos un heurístico basado en la comprobación de la existencia de la matriz inversa de la aplicación lineal $D_x(f)$ en varios puntos x tomados al azar. Dicho heurístico se muestra efectivo empíricamente para generar sistemas válidos hasta polinomios en 10 variables¹.

Para crear dichos slp 's reutilizaremos el mecanismo usado para la generación de la población inicial descrito en la Subsección 8.2.1. Posteriormente aplicaremos todos los algoritmos sobre el mismo sistema.

A continuación presentamos los parámetros usados en cada uno de los algoritmos evolutivos:

Cuadro 12.1: *Parámetros algoritmo evolutivo genético para la resolución del problema XVII de Smale.*

Probabilidad de Cruce:	0.9
Probabilidad de Mutación:	0.05
Tamaño de la Población:	100
Número de Generaciones:	100
Operador de Cruce:	Aritmético
Operador de Mutación:	No uniforme
Operador de Selección:	Simple
Operador de Reemplazo:	Por ranqueo
Generador de números aleatorios:	Propio del Sistema Operativo

¹En las pruebas usando más de 10 variables hemos tenido que *relajar* dicho heurístico así como realizar las pruebas generando solo 100 sistemas. Aún así, las pruebas con este número de variables, se alargaron durante varios días.

Cuadro 12.2: *Parámetros algoritmo evolutivo diferencial para la resolución del problema XVII de Smale.*

Probabilidad de Cruce:	0.9
Probabilidad de Mutación:	0.05
Tamaño de la Población:	100
Número de Generaciones:	100
Operador de Cruce:	Diferencial
Operador de Mutación:	No uniforme
Operador de Selección:	Simple
Operador de Reemplazo:	Por ranqueo
Generador de números aleatorios:	Propio del Sistema Operativo

Cuadro 12.3: *Parámetros algoritmo evolutivo basado en enfriamiento simulado para la resolución del problema XVII de Smale.*

Probabilidad de Cruce:	0.05
Probabilidad de Mutación:	0.9
Tamaño de la Población:	100
Número de Generaciones:	100
Operador de Cruce:	Aritmético
Operador de Mutación:	No uniforme
Operador de Selección:	Simple
Operador de Reemplazo:	Enfriamiento simulado
Probabilidad de Reemplazo:	0.9
Generador de números aleatorios:	Propio del Sistema Operativo

Cuadro 12.4: *Parámetros algoritmo evolutivo basado en enjambres de partículas para la resolución del problema XVII de Smale.*

Probabilidad de Cruce:	0.9
Probabilidad de Mutación:	0.05
Tamaño de la Población:	100
Número de Generaciones:	100
Operador de Cruce:	Enjambre
Operador de Mutación:	No uniforme
Operador de Selección:	Simple
Operador de Reemplazo:	Por ranqueo
Probabilidad de Reemplazo:	0.9
Generador de números aleatorios:	Propio del Sistema Operativo

Los resultados que mostraremos serán:

Número de Bézout (\mathcal{D}): Media de los números de Bézout de los sistemas.

Longitud del hipercubo (H): Media de la longitud del hipercubo donde es altamente probable que se encuentren las raíces (véase Corolario 0.0.34).

Distancia entre raíces (h): Media de la distancia mínima entre raíces de los sistemas (véase la Nota 0.0.19).

Tiempo de ejecución (t): Duración media, en segundos, de la prueba.

Tasa de éxito (t_e): Porcentaje de sistemas en los que finalmente se ha encontrado una solución.

Nótese que los tres primeros son comunes a todos los métodos propuestos pues son propiedades de los sistemas de ecuaciones polinomiales test mientras que los dos últimos dependen del algoritmo usado. Mostraremos los resultados agrupados por algoritmo y método de calificación usado.

12.4.2 Resultados experimentales

Detallamos a continuación los resultados experimentales obtenidos al ejecutar el experimento anterior.

Cuadro 12.5: *Parámetros descriptivos de los sistemas de ecuaciones.*

n	\mathcal{D}	H	h
3	18.78	987.46	$4.18 \cdot 10^{-6}$
5	328.02	17,247.4	$1.07 \cdot 10^{-7}$
7	6,066.85	$3.19 \cdot 10^5$	$7.31 \cdot 10^{-9}$
10	$3.21 \cdot 10^5$	$1.69 \cdot 10^7$	$2.1 \cdot 10^{-9}$
11	$4.69 \cdot 10^5$	$2.46 \cdot 10^7$	$1.79 \cdot 10^{-9}$
12	$2.42 \cdot 10^6$	$1.27 \cdot 10^8$	$1.45 \cdot 10^{-9}$

Cuadro 12.6: *Resultados algoritmos evolutivos genéticos.*

(a) Calificación α^* .			(b) Calificación α_c^* .		
n	t (s)	t_e	n	t (s)	t_e
3	0.33	60.33 %	3	0.98	82.33 %
5	0.68	13.33 %	5	3.14	49.67 %
7	0.96	2 %	7	13.55	20.67 %
10	1.74	0.67 %	10	49.86	4.67 %
11	1.88	0 %	11	56.92	6 %
12	1.99	2 %	12	66.67	7 %

Cuadro 12.7: *Resultados algoritmos evolutivos diferenciales.*

(a) Calificación α^* .			(b) Calificación α_c^* .		
n	t (s)	t_e	n	t (s)	t_e
3	0.32	60.67 %	3	0.94	87.67 %
5	0.65	13.33 %	5	3.07	87.33 %
7	0.89	1.67 %	7	12.59	70.33 %
10	1.55	1 %	10	47.05	25.33 %
11	1.64	0 %	11	50.81	29 %
12	1.84	2 %	12	59.84	28 %

Cuadro 12.8: *Resultados algoritmos de enfriamiento simulado.*

(a) Calificación α^* .			(b) Calificación α_c^* .		
n	t (s)	t_e	n	t (s)	t_e
3	0.32	59.33 %	3	0.95	86.67 %
5	0.66	15.67 %	5	2.78	82.33 %
7	0.92	0.67 %	7	10.98	71.67 %
10	1.66	1 %	10	45.14	27.67 %
11	1.78	0 %	11	50.02	32 %
12	1.91	2 %	12	60.33	30 %

Cuadro 12.9: *Resultados algoritmos de enjambre de partículas.*

(a) Calificación α^* .			(b) Calificación α_c^* .		
n	t (s)	t_e	n	t (s)	t_e
3	0.2	77.67 %	3	0.9	88 %
5	0.58	67.33 %	5	1.43	94 %
7	0.98	63.33 %	7	3.88	94.67 %
10	1.75	39.67 %	10	19.93	85.33 %
11	1.89	43 %	11	20.4	91 %
12	2.14	42 %	12	30.51	84 %

A partir de los resultados podemos inferir algunas conclusiones lógicas:

- El tiempo de ejecución aumenta con el aumento del número de variables. Cuando usamos la calificación α^* aproximadamente de forma lineal mientras que cuando usamos la calificación α_c^* aproximadamente de forma cuadrática.
- Por el contrario, la tasa de éxito disminuye con el aumento del número de variables. Muy rápidamente en el caso de usar la calificación α^* y más lentamente cuando usamos la calificación α_c^* . La excepción a este comportamiento la encontramos en el

algoritmo basado en enjambre de partículas pues en ambos casos disminuye lentamente, de hecho en el caso de usar la calificación α_c^* aumenta ligeramente para 5 y 7 variables.

Nótese que los resultados para más de 10 variables se refieren a 100 repeticiones, con lo cual el margen de incertidumbre aumenta.

- La tasa de éxito es muy superior en el caso de usar el mecanismo de calificación α_c^* en todos los casos y algoritmos.
- El algoritmo basado en enjambre de partículas es muy superior al resto en ambos casos. No solo su tasa de éxito es mayor, sino que desciende mucho más lentamente y además es mucho más rápido.

Nota 12.4.1. *Nótese que los tamaños de los retículos que empíricamente hemos obtenido es de orden $\mathcal{O}(10^{n+6})$ aproximadamente. Si quisiéramos ejecutar el algoritmo exhaustivo, suponiendo que tenemos un ordenador capaz de ejecutar 10^4 comprobaciones por segundo, tardaríamos un día en resolver el problema para 3 variables. Para el de 7 variables tardaríamos alrededor de 31 años.*

El uso del algoritmo exhaustivo con refinamiento de retículo, en el mejor de los casos, nos evitaría el cálculo de la mitad de los test. Por consiguiente, el algoritmo sigue siendo impracticable.

Nota 12.4.2. *Por completitud comentamos brevemente los resultados de ejecutar los métodos implementados para resolver este tipo de problemas en el software comercial MAPLETM tanto simbólicos (basados en bases de Gröbner) como numéricos (basados en algoritmos de deformación homotópica similares a los explicados en la introducción, salvo que en este caso los puntos iniciales son tomados siguiendo estrategias heurísticas híbridas simbólico-numéricas):*

Para el ejemplo de 7 ecuaciones con 7 incógnitas, el algoritmo simbólico no pudo ser ejecutado pues excedía la capacidad de cálculo del ordenador usado. Sin embargo, el algoritmo numérico tardó alrededor de un segundo en obtener una solución, con lo cual estamos en el mismo rango de tiempos.

Téngase en cuenta que dichos algoritmos esta fuertemente optimizados para la utilización de varios procesadores. Los algoritmos presentados en esta memoria también son susceptibles de optimizar en ese mismo sentido, pero actualmente no han sido programados de dicha forma.

Apéndice A

Las funciones Gamma y Beta

*Lisez Euler, lisez Euler, c'est notre maître à tous*¹.
Pierre-Simon Laplace [Lib46, pág 51]

En este anexo introduciremos las propiedades fundamentales de las funciones Γ y β que se han usado en múltiples ocasiones a lo largo de la memoria. Ambas funciones pertenecen al difuso conjunto de las *funciones especiales* y fueron estudiadas principalmente por Euler y Legendre durante el siglo XVIII y posteriormente por Gauß y Weierstraß. La idea principal era extender las nociones de factorial de un número natural y de coeficiente binomial de dos números naturales a argumentos complejos.

A.1 Historia de las funciones Γ y β

La historia de estas dos funciones comienza en el siglo XVIII cuando Daniel Bernoulli y Christian Goldbach plantean el problema de extender el factorial a valores no enteros. La primera extensión la presenta Euler en 1720:

$$n! = \prod_{k=1}^{\infty} \frac{\left(1 + \frac{1}{k}\right)^n}{1 + \frac{1}{k}}.$$

10 años después comunica a Goldbach la siguiente igualdad

$$n! = \int_0^1 (-\log s)^n ds,$$

válida para $n > 0$, que mediante el cambio de variable $t = -\log s$ obtenemos la actual definición:

Definición A.1.1 (Función Γ o *integral de Euler de segunda especie*). *Definimos la función Γ como el resultado de la siguiente integral:*

$$\Gamma(z) := \int_0^{\infty} t^{z-1} e^{-t} dt,$$

válida en todo número complejo z tal que su parte real $\Re(z)$ sea mayor que 0.

¹Leed a Euler, leed a Euler, él es nuestro maestro en todo.

Esta función se puede extender analíticamente a todo el plano complejo excepto los números enteros negativos, en donde la función Γ presenta polos simples.

A lo largo del siglo XIX se suceden los descubrimientos alrededor de esta función. Gauß proporciona la siguiente igualdad:

$$\Gamma(z) = \lim_{m \rightarrow \infty} \frac{m^z m!}{z(z+1)(z+2) \cdots (z+m)},$$

válida para todo número complejo excepto los números enteros negativos. Posteriormente Karl Weierstraß introduce la siguiente igualdad:

$$\Gamma(z) = \frac{e^{-\gamma z}}{z} \prod_{k=1}^{\infty} \left(1 + \frac{z}{k}\right)^{-1} e^{\frac{z}{k}},$$

a partir de la cual derivó el conocido *Teorema de Factorización de Weierstraß* y por consiguiente el *Teorema Fundamental del Álgebra*.

Finalmente, ya en el siglo XX se consigue dar una caracterización de la función Γ .

Teorema A.1.2 (Teorema de Bohr-Mollerup). *La función Γ es la única función, en el intervalo $x > 0$, que simultáneamente cumple:*

- $\Gamma(1) = 1$,
- $\Gamma(x+1) = x\Gamma(x)$ para todo $x > 0$ y
- $\log \Gamma$ es una función logarítmicamente convexa.

Sin embargo, para el caso del coeficiente binomial, la historia es bastante más sencilla. De forma prácticamente trivial podemos extender su definición a todo el plano complejo para el primer parámetro usando la siguiente definición:

$$\binom{z}{n} := \frac{z(z-1)(z-2) \cdots (z-n+1)}{n(n-1)(n-2) \cdots 1}.$$

Dicha función está bien definida para todo valor z perteneciente a los números complejos y para todo valor n perteneciente a los números naturales. Además, claramente se observa que en el caso que z sea un número natural recuperamos la definición estándar del coeficiente binomial.

Esta generalización permite generalizar la fórmula binomial, fijando una de las variables a 1. Es decir

$$(1+X)^z = \sum_{n=0}^{\infty} \binom{z}{n} X^n$$

es válida para todos los números complejos z y X con $|X| < 1$ y sigue cumpliendo las buenas propiedades. Principalmente:

$$(1+X)^z(1+X)^t = (1+X)^{z+t} \text{ y } ((1+X)^z)^t = (1+X)^{zt}.$$

Ahora bien, la generalización sobre sus dos argumentos como números complejos es más complicada y fue estudiada por Euler y Legendre. Una posible idea es partir de la fórmula factorial del número combinatorio sustituyendo las factoriales por la función Γ vista anteriormente. Es decir:

$$\binom{x}{y} = \frac{\Gamma(x+1)}{\Gamma(y+1)\Gamma(x-y+1)}. \quad (\text{A.1})$$

De esta forma habremos extendido la noción de número combinatorio a todos los valores en los que está definido la función Γ ya que dicha función nunca toma el valor 0. Sin embargo dicha generalización presenta diversos problemas:

- No está muy bien estudiada.
- Algunas identidades que cumplen los coeficiente binomiales dejan de ser válidas en esta generalización.
- El comportamiento de dicha función es bastante errático.

Sin embargo, existe otra generalización que tiene mejores propiedades, es la llamada función β .

Definición A.1.3 (Función β o *integral de Euler de primera especie*). *Definimos la función β como el resultado de la siguiente integral:*

$$\beta(x, y) := \int_0^1 t^{x-1}(1-t)^{y-1} dt,$$

la cual es válida para todos los números complejos x e y tales que su parte real $\Re(x)$ y $\Re(y)$ sea mayor que 0.

Su principal propiedad es su relación íntima con la función Γ según la siguiente fórmula.

$$\beta(x, y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}. \quad (\text{A.2})$$

A partir de la cual es inmediato encontrar una mejor generalización de los coeficiente binomiales usando las Ecuaciones (A.1) y (A.2):

$$\binom{x}{y} = \frac{1}{(x+1)\beta(x-y+1, y+1)}.$$

En la actualidad ambas funciones aparecen principalmente como distribuciones de probabilidad y combinatoria aunque sorprendentemente aparece en algunos invariantes de tipo geométrico como ya hemos visto a lo largo de la memoria.

A.2 Propiedades de las funciones Γ y β

La principal propiedad que posee la función Γ es la llamada *fórmula recursiva*:

$$\Gamma(z + 1) = z\Gamma(z)$$

válida en todo el dominio de la función. A partir de dicha propiedad se deduce que para todo número natural n se tiene:

$$\Gamma(n) = (n - 1)!$$

Otras propiedades interesantes son:

Fórmula de reflexión de Euler:

$$\Gamma(z)\Gamma(z - 1) = \frac{\pi}{\sin(\pi z)}.$$

Fórmula de la duplicación de Legendre:

$$\Gamma(2z) = \pi^{-\frac{1}{2}} 2^{2z-1} \Gamma(z) \Gamma\left(z + \frac{1}{2}\right)$$

para valores de z tales que $2z \notin \{0, -1, -2, \dots\}$.

Fórmula de multiplicativa de Gauß:

$$\Gamma(nz) = (2\pi)^{\frac{1-n}{2}} n^{nz-\frac{1}{2}} \prod_{k=0}^{n-1} \Gamma\left(z + \frac{k}{n}\right)$$

para valores de z tales que $nz \notin \{0, -1, -2, \dots\}$.

A partir de dichas propiedades se demuestra fácilmente la siguiente proposición.

Proposición A.2.1. *Conociendo los valores de la función Γ en el intervalo $[0, 1]$ se obtienen los valores de toda la recta real.*

Demostración. Para todo número real x . Supongamos que $x > 0$, entonces existe un número natural n y ε en el intervalo $[0, 1]$ tal que $x = n + \varepsilon$. Usando la fórmula recursiva se tiene:

$$\Gamma(n + \varepsilon) = (n - 1 + \varepsilon)\Gamma(n - 1 + \varepsilon) = \dots = (n - 1 + \varepsilon)(n - 2 + \varepsilon) \dots \varepsilon\Gamma(\varepsilon).$$

En el caso que $x < 0$ tenemos que existe un número natural n y ε en el intervalo $[0, 1]$ tal que $x = -n + \varepsilon$. Usando la fórmula de reflexión de Euler obtenemos:

$$\Gamma(-n + \varepsilon)\Gamma(n + 1 - \varepsilon) = \frac{\pi}{\sin(\pi(1 + n - \varepsilon))}$$

que podemos calcular $\Gamma(n + 1 - \varepsilon)$ como el caso anterior y despejando completamos la demostración. \square

Finalmente, posiblemente el valor más usado de la función Γ sea el dado por la igualdad

$$\Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}$$

que se puede obtener directamente de la definición simplemente realizando el cambio de variable $u = \sqrt{t}$.

En el caso de la función β su principal propiedad, aparte de la ya mencionada en la (A.2), es ser simétrica. También son interesantes las siguientes definiciones alternativas de dicha función:

$$\beta(x, y) = 2 \int_0^{\frac{\pi}{2}} (\sin t)^{2x-1} (\cos t)^{2y-1} dt,$$

$$\beta(x, y) = \int_0^{\infty} \frac{t^{x-1}}{(1+t)^{x+y}} dt.$$

Así como la identidad:

$$\beta(x, y)\beta(x+y, 1-y) = \frac{\pi}{x \sin(\pi y)}.$$

Las mejores aproximaciones a ambas funciones se consiguen mediante la llamada fórmula de Stirling.

$$\Gamma(z) \approx \sqrt{\frac{2\pi}{z}} \left(\frac{z}{e}\right)^z.$$

$$\beta(x, y) \approx \sqrt{2\pi} \frac{x^{x-\frac{1}{2}} y^{y-\frac{1}{2}}}{(x+y)^{x+y-\frac{1}{2}}}.$$

Asimismo, una forma eficiente de calcular valores es mediante cuadraturas obtenidas directamente sobre la definición.

Apéndice B

Protocolo de experimentación

To call in the statistician after the experiment is done may be no more than asking him to perform a postmortem examination: he may be able to say what the experiment died of¹.

Sir Ronald Fisher [Fis38]

A lo largo de las siguientes páginas detallaremos el protocolo que hemos seguido para la realización de las pruebas experimentales. Para la realización de dichas pruebas hemos usado un software propio, escrito en C++, en el que hemos implementado tanto los algoritmos evolutivos y de programación genética como las estructuras de datos *slp* y árbol, así como los operadores genéticos asociados a dichas estructuras.

Este software se ha ejecutado en una estación de trabajo Dell Precision T5500 que cuenta con 12 GB de memoria RAM y un procesador Intel Xeon E5520 de cuádruple núcleo, todo ello gestionado por un entorno Gentoo Linux.

B.1 Selección del concepto a aprender

El primer paso del protocolo es la generación de una muestra aleatoria simple de conceptos. Para ello es necesario fijar una distribución de probabilidad en el espacio de conceptos que en una primera aproximación podríamos tomar como el espacio de todas las funciones infinitamente diferenciables. Ahora bien, dicho espacio de funciones tiene dimensión infinita (además no numerable) y por lo tanto es extremadamente complejo tomar una distribución de probabilidad en dicho espacio que además sea susceptible de ser usada a efectos prácticos. Es por ello que decidimos realizar dos reducciones distintas del problema.

Tomaremos como espacio de conceptos:

- El espacio $P_n^{\mathbb{R}}[X]$ de polinomios univariados de grado acotado por n .

¹Llamar al estadístico después de que el experimento haya finalizado es lo más o menos lo mismo que solicitarle que realice un examen *post-mortem*: en el mejor de los casos es capaz de explicar de que murió el experimento.

- El espacio $SLP_l(\mathcal{F}, \mathcal{T})$ de *slp's* longitud acotada por l con conjunto de funciones $\mathcal{F} := \{+, -, *, \sqrt{}, \sin, \cos, \ln, \exp\}$ y conjunto de terminales $\mathcal{T} := \{x, c\}$.

Nótese que en un sentido estricto las únicas funciones computables son precisamente las que se pueden expresar como un elemento en alguno de los espacios anteriormente citados.

Otra forma de encarar el anterior problema hubiera sido usando un *proceso de Gauß* tal y como se explica en [Ras06] (véase también [Cre93]).

Pasamos a detallar el proceso de generación de las muestras en ambos casos.

B.1.1 Polinomios de grado acotado

En este caso tenemos que el espacio de conceptos es un espacio vectorial de dimensión finita, por lo que, no solo el modelo es conocido sino que, además, es lineal. En este caso utilizar la programación genética para resolver un problema de regresión simbólica es ciertamente ineficiente pues cualquier variante de la llamada *regresión lineal* produce resultados óptimos en mucho menor tiempo. Aún así consideramos que es un buen ejercicio comprobar el desempeño de nuestros métodos en un entorno predecible y controlable.

La literatura acerca de este tema es extensa (véase por ejemplo los trabajos de Kostlan [Kos93, EK95]) y está íntimamente relacionada con el estudio de las matrices aleatorias. Como estamos tratando con un espacio vectorial de dimensión finita y, por lo tanto, no compacto no es posible tomar la distribución uniforme. En su defecto usaremos la clásica normal $\mathcal{N}(0, 1)$ centrada en 0 y con varianza 1. Esta elección no es arbitraria pues dicha distribución de probabilidad es equiprobable a la distribución de probabilidad uniforme en el espacio proyectivo de polinomios $H_n^{\mathbb{R}}[X]$ que es la compactificación natural del espacio $P_n^{\mathbb{R}}[X]$.

Ahora bien, en este espacio es bien sencillo general una muestra aleatoria simple. Una vez fijado el grado máximo n solamente es necesario general los $n + 1$ coeficientes del polinomio siguiendo cada uno de ellos una distribución normal $\mathcal{N}(0, 1)$.

B.1.2 *Slp's* de longitud acotada

En este caso tenemos que el modelo es completamente desconocido por lo que es procedente usar la programación genética. Ahora bien, es cierto que el modelo es «representable» en el espacio de búsqueda lo cual no tiene que ser cierto en general.

Otro problema que existe con esta forma de general el concepto es el desconocimiento general en la relación que hay entre la distribución de probabilidad en el espacio de parámetros de un *slp* y la distribución de probabilidad en el espacio de aplicación semántica que produce. Aún con este inconveniente usaremos este procedimiento.

En una primera aproximación parece razonable usar la distribución uniforme en el espacio de parámetros para la generación del concepto, es decir, usar el mismo procedimiento que hemos usado en la Subsección 8.2.1. Sin embargo, este procedimiento produce un gran número de *slp's* de longitud efectiva muy baja y por lo tanto en multitud de ocasiones constantes o lineales. Es por ello que preferimos usar la distribución de probabilidad

inducida por el procedimiento usado para generar los árboles mostrada en la introducción. Nótese que los árboles así generados son un subconjunto del conjunto $SLP_l(\mathcal{F}, \mathcal{T})$ y por lo tanto la distribución de probabilidad se extiende de forma natural al conjunto $SLP_l(\mathcal{F}, \mathcal{T})$.

B.2 Generación de una muestra aleatoria simple

Una vez escogido el concepto f a aprender es necesario generar dos muestras:

- Una muestra de aprendizaje de 30 elementos, que puede contener ruido Gausiano.
- Una muestra de validación de 200 elementos, que *nunca* contendrá ruido.

En ambos casos la muestra se generará mediante puntos x_i uniformemente distribuidos en un intervalo de longitud finita fijado de antemano. Una vez escogido dichos puntos almacenamos los valores $(x_i, y_i = f(x_i))$. Salvo que se diga lo contrario, ninguna de las muestras contendrá ruido.

La muestra de aprendizaje será usada para evolucionar nuestra población de *slp's* mientras que la muestra de validación solo su usará para comprobar el desempeño de nuestro algoritmo.

B.3 Ejecución de las pruebas

Una vez generada las muestras ejecutamos el programa tantas veces como valores del parámetro sujeto de estudio se tenga. A esto lo llamaremos una *ronda* de aprendizaje. Las muestras no se modifican entre las ejecuciones del programa dentro de una ronda aunque si que se modifica la semilla aleatoria.

Con el fin de obtener resultados estadísticamente válidos realizaremos 300 rondas. Ahora bien, en cada ronda se generarán nuevas muestras. A este procedimiento lo llamaremos *experimento*. Así un experimento esta compuesto de 300 rondas, todas ellas con muestras independientes.

Una vez finalizada la ejecución procederemos a la validación. Para ello se extrae el mejor elemento, que denominaremos *élite*, de la última población y se calcula su *MSE* sobre la muestra de validación que, recordemos, es una muestra de 200 puntos que no contiene errores y que no se ha usado en la fase de aprendizaje (véase por ejemplo [CR09] para una discusión en este sentido).

Detallamos ahora los parámetros comunes a todos los experimentos que se han realizado:

Cuadro B.1: *Parámetros comunes.*

Longitud de los $slp's$:	16
Probabilidad de Cruce:	0.9
Probabilidad de Mutación:	0.05
Tamaño de la Población:	200
Número máximo de operaciones:	10^7
Operador de Selección en la programación genética:	Simple
Operador de Selección en el algoritmo evolutivo:	2-Torneo
Operador de Reemplazo:	Por ranqueo
Generador de números aleatorios:	Propio del Sistema Operativo

Nótese que en este contexto el número de operaciones es el número de variables no terminales que han sido evaluadas.

Así mismo los parámetros comunes en los experimentos usando las muestras polinómicas serán:

Cuadro B.2: *Parámetros comunes muestras polinómicas.*

Conjunto de terminales:	$\mathcal{T} := \{x, c_1, c_2, 1\}$ donde $x, c_1, c_2 \in [-1, 1]$.
Conjunto de funciones:	$\mathcal{F} := \{+, -, *, /, \sqrt{\cdot}\}$.
Grado máximo del polinomio generador:	5.

Donde x denota la variable y c_1, c_2 son dos parámetros que se generan aleatoriamente al generar la muestra. En los experimentos realizados en el Capítulo 10 son dichos parámetros los que serán optimizados.

Finalmente el resto de parámetros comunes en los experimentos usando las muestras que proceden de los $slp's$ aleatorios serán:

Cuadro B.3: *Parámetros comunes muestras $slp's$.*

Conjunto de terminales:	$\mathcal{T} := \{x, c\}$ con $x \in [-1, 1]$ y $c \in [-1, 1]$.
Conjunto de funciones:	$\mathcal{F} := \{+, -, *, \sqrt{\cdot}, \sin, \cos, \ln, \exp\}$.
Longitud máxima del slp generador:	16.
Conjunto de terminales slp generador:	$\mathcal{T} := \{x, c\}$ con $x \in [-1, 1]$ y $c \in [-1, 1]$.
Conjunto de funciones slp generador:	$\mathcal{F} := \{+, -, *, \sqrt{\cdot}, \sin, \cos, \ln, \exp\}$.

Donde x denota la variable y c denota un parámetro como comentamos en el párrafo anterior.

Hay que puntualizar que en todas las ejecuciones se ha activado el elitismo. Esto significa que al finalizar cada generación se guarda el mejor elemento y éste reemplaza al peor elemento de la siguiente generación, siempre y cuando el valor guardado sea mejor. De esta forma nos aseguramos que al finalizar cada generación la calificación del mejor elemento de la población no empeora.

B.4 Resultados que se muestran

Para cada experimento y tipo de prueba mostramos los siguientes resultados:

Tabla con la tasa de éxito y la tasa de error: Diremos que una ejecución ha sido *errónea* si la validación del modelo es superior al 150 % del rango intercuartílico. Es decir, si la validación cumple:

$$\mathcal{Val}(\Gamma) \geq 1.5|Q_3 - Q_1|$$

donde Q_1, Q_3 representa el valor del primer y tercer cuartil respectivamente de la prueba con dicho parámetro. Extraeremos dichos elementos de la muestra pues distorsionan enormemente los resultados.

Diremos que una ejecución ha sido *exitosa* si la validación del modelo es inferior al 10 % del rango de la muestra. Es decir si la validación cumple:

$$\mathcal{Val}(\Gamma) \leq 0.1 \left| \max_{1 \leq i \leq 30} y_i - \min_{1 \leq i \leq 30} y_i \right|$$

donde Γ representa el modelo escogido por el algoritmo y y_i es la muestra de aprendizaje. Nótese que los cálculos sobre la tasa de éxito están referidos a la muestra sin los resultados erróneos.

Como curiosidad, el autor de esta memoria ha visto resultados en los cuales sólo existía **un** valor erróneo que era 4 órdenes de magnitud superior al siguiente peor resultado. Obviamente todos los estadísticos estaban dominados por ese único resultado erróneo lo cual no parece muy razonable. Téngase en cuenta que estamos tratando con un algoritmo probabilista cuya probabilidad de error no está acotada, o por lo menos el autor no conoce ningún resultado en este sentido. La tasa de error podría servir como una aproximación a dicha probabilidad.

Tablas con un resumen estadístico descriptivo: En la primera tabla se presentan la media y varianza de los resultados de validación. En la segunda, se muestran los valores máximo y mínimo, la mediana así como los cuartiles primero y tercero de la validación. En todos los casos, estos estadísticos se refieren a la muestra en la que se han extraído previamente las elementos considerados erróneos en la anterior tabla.

Diagrama de cajas y bigotes: Forma gráfica en el que se representan los valores numéricos de la anterior tabla.

En este diagrama se muestra una caja y dos asas (bigotes). El extremo inferior de la caja representa el valor del primer cuartil (q_1 en la tabla anterior) de las calificaciones de validación mientras que el extremo superior de la caja representa el tercer cuartil (q_3). La raya gruesa dentro de la caja representa la mediana (q_2). Si las muescas que tiene una caja no se solapan con las de otra indica que, con alta probabilidad, las medianas de las distribuciones de probabilidad son distintas (véase [CCKT83] para más detalles).

Los dos bigotes representan los valores mínimos y máximos de la muestra a la que previamente se le han extraído los errores.

Nótese que un conjunto de parámetros es mejor cuanto más estrecha sea la caja (menos varianza presenta) y más abajo se encuentre (mejores individuos produce).

Tabla con los resultado de los contrastes de hipótesis cruzados: En la columna (i, j) se muestra el *p-valor* asociado a un contraste de hipótesis no paramétrico mediante el test para dos muestras de Kolmogorov-Smirnov. Nuestra hipótesis nula H_0 será que, la función de distribución empírica (*ecdf*) de la validación para el valor del parámetro sujeto de estudio en la fila i no es mayor que la *ecdf* para el valor del parámetro en la columna j .

De esta forma, si el valor de la casilla (i, j) es menor que el valor de la significancia α escogido, podemos rechazar la hipótesis nula. Es decir la *ecdf* es estrictamente mayor y por lo tanto dicho valor del parámetro ofrece un mejor rendimiento sobre dicha muestra pues obtendrá, con mayor probabilidad, modelos con una calificación de validación inferior. Por norma general tomaremos como valor de la significancia $\alpha = 0.1$. Nótese que los valores superiores a 0.1 del *p-valor* deben tomarse como > 0.1 , no como su valor numérico (tal y como indica el manual del programa usado para realizar los cálculos estadísticos [Tea]).

Así mismo, nótese que los resultados pueden ser aproximados en algunos casos (sobre todo en el caso de las muestras generadas mediante *slp's* aleatorios) pues la distribución de probabilidad debe (y de hecho tiene) al menos un punto con probabilidad, el 0. Por lo que la distribución de probabilidad asociada a la validación no es continua. Es decir, la probabilidad de que el algoritmo obtenga la función **exacta** con la que se generó la muestra es distinta de cero, tal como se podría esperar si el algoritmo funciona correctamente.

Nótese que, como hemos hecho previamente, consideramos sólo la muestra sin errores.

Bibliografía

- [AB00] D.A. Augusto and H. Barbosa, *Symbolic regression via genetic programming*, 6th Brazilian Symposium on Neural Networks (SBRN 2000), 22-25 November 2000, Rio de Janeiro, Brazil (Los Alamitos, CA, USA), IEEE Computer Society, 2000, pp. 173–178.
- [Abe26] N.H. Abel, *Beweis der unmöglichkeit, algebraische gleichungen von höheren graden, als dem vierten, allgemein aufzulösen*, Journal für die reine und angewandte Mathematik **1** (1826), 65–97, <http://gdz.sub.uni-goettingen.de/dms/load/img/>.
- [Aka70] H. Akaike, *Statistical Prediction Information*, Ann. Inst. Statistic. Math **22** (1970), 203–217.
- [Aka74] ———, *A new look at the statistical model identification*, IEEE Transactions on Automatic Control **19** (1974), no. 6, 716–723.
- [AMB09] C.L. Alonso, J.L. Montaña, and C.E. Borges, *Evolution Strategies for Constants Optimization in Genetic Programming.*, ICTAI 2009: 21st IEEE International Conference on Tools with Artificial Intelligence, IEEE Computer Society, 2009, pp. 702–707.
- [AMPB09] C.L. Alonso, J.L. Montaña, J. Puente, and C.E. Borges, *A New Linear Genetic Programming Approach Based on Straight Line Programs: Some Theoretical and Exeperimental Aspects.*, International Journal on Artificial Intelligence Tools **18** (2009), no. 5, 757–781.
- [Ati02] M. Atiyah, *Mathematics in the 20th century*, Bulletin of the London Mathematical Society **34** (2002), no. 1, 1–15.
- [AW05] J. Azaïs and M. Wschebor, *On the Roots of a Random System of Equations. The Theorem on Shub and Smale and Some Extensions*, Found. Comput. Math. **2** (2005), 125–144.
- [BAdlP+10] C.E. Borges, C.L. Alonso, A.O. de la Puente, M.C. Echeandia, and J.L. Montaña, *Coevolutionary architectures with straight line programs for solving the symbolic regression problem.*, Proceedings of the International Conference on Evolutionary Computation 2010, ICEC, 2010.

- [BAM10] C.E. Borges, C.L. Alonso, and J.L. Montaña, *Model selection in genetic programming*, Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation Conference, ACM Proceedings, ACM, 2010, pp. 985–986.
- [Ban93] W. Banzhaf, *Genetic programming for pedestrians*, Proceedings of the 5th International Conference on Genetic Algorithms (San Francisco, CA, USA), Morgan Kaufmann Publishers Inc., 1993, pp. 628–639.
- [BB69] N. Balabanian and T.A. Bickart, *Electrical network theory*, John Wiley and Sons, Inc., 1969.
- [BB07] W. Banzhaf and M. Brameier, *Linear Genetic Programming*, Springer, 2007.
- [BBEM90] B. Beauzamy, E. Bombieri, P. Enflo, and H.L. Montgomery, *Products of polynomials in many variables*, Journal of Number Theory **36** (1990), no. 2, 219–245.
- [BCL06] P. Bürgisser, F. Cucker, and M. Lotz, *Smoothed Analysis of Complex Conic Condition Numbers*, Journal de Mathématiques Pures et Appliquées (2006), 293–309.
- [BCR87] J. Bochnak, M. Coste, and M-F. Roy, *Géométrie Algébrique Réelle*, Springer-Verlag, Berlin, 1987.
- [BCSS98] L. Blum, F. Cucker, M. Shub, and S. Smale, *Complexity and Real Computation*, Springer-Verlag, New York, 1998.
- [Bel] C. Beltrán, *Certified Numerical Homotopy Tracking*, Sometido.
- [Bel06] ———, *Sobre el problema XVII de Smale: Teoría de la intersección y geometría integral*, Ph.D. thesis, Universidad de Cantabria, 2006.
- [Béz79] E. Bézout, *Théorie générale des équations algébriques*, Ph.D. thesis, Académie Royale des Sciences & de celle de la Marine, 1779.
- [BP] C. Beltrán and L.M. Pardo, *Fast linear homotopy to find approximate zeros of polynomial systems*, Sometido.
- [BP07a] ———, *Estimates on the Distribution of the Condition Number of Singular Matrices*, Found. Comput. Math. **7** (2007), no. 1, 87–134.
- [BP07b] ———, *On the Probability Distribution of Condition Numbers of Complete System Varieties and the Average Radius of Convergence of Newton Method in the Under-determined Case*, Math. Comp. **76** (2007), 1393–1424.
- [BP08a] ———, *On Smale’s 17 Problem: A Probabilistic Positive Solution*, Found. Comput. Math. **8** (2008), no. 1, 1–43.
- [BP08b] C.E. Borges and L.M. Pardo, *On the Probability Distribution of Data at Points in Real Complete Intersection Varieties*, Journal of Complexity **24** (2008), no. 4, 492–523.

-
- [BP09] C. Beltrán and L.M. Pardo, *On Smale's 17 Problem: Average Polynomial Solver for Affine and Projective Solutions*, J.Amer. Math. Soc. **22** (2009), 363–385.
- [BS94] J. Bernardo and A. F. M. Smith, *Bayesian Theory*, John Wiley & Sons, 1994.
- [BS09] C. Beltrán and M. Shub, *Complexity of Bezout's Theorem VII: Distance Estimates in the Condition Metric*, Foundations of Computational Mathematics **9** (2009), no. 2, 179–195.
- [BSS89] L. Blum, M. Shub, and S. Smale, *On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines*, Bull. Amer. Math. Soc. (N.S.) **21** (1989), no. 1, 1–46.
- [BT97] D. Bau and L.N. Trefethen, *Numerical Linear Algebra*, SIAM, Philadelphia, 1997.
- [BW93] T. Becker and V. Weispfenning, *Gröbner Bases, Graduate Texts in Mathematics*, Springer-Verlag, New York, 1993.
- [Car45] G. Cardano, *Artis Magnæ, Sive de Regulis Algebraicis Liber Unus*, Joh. Petreium, Nürnberg, 1545.
- [CAW03] J.A. Cuesta-Albertos and M. Wschebor, *Some Remarks on the Condition Number of a Real Random Square Matrix*, J.Complexity **19** (2003), 548–554.
- [CCKT83] J.M. Chambers, W.S. Cleveland, B. Kleiner, and P.A. Tukey, *Graphical Methods for Data Analysis*, Chapman and Hall, New York, 1983.
- [CD05] Z. Chen and J.J. Dongarra, *Condition Numbers of Gaussian Random Matrices*, J. Matrix Anal. Appl. SIAM **27** (2005), 603–620.
- [CGH⁺03] D. Castro, M. Giusti, J. Heintz, G. Matera, and L. M. Pardo, *The hardness of polynomial equation solving*, Foundations of Computational Mathematics **3** (2003), no. 4, 347–420.
- [Cha08] K.U. Chakraborty, *Advances in Differential Evolution*, Springer Publishing Company, Incorporated, 2008.
- [Chu36] A. Church, *An unsolvable problem of elementary number theory*, American Journal of Mathematics **58** (1936), 345–363.
- [CJW06] J. Carlson, A. Jaffe, and A. Wiles (eds.), *The Millennium Prize Problems*, Clay Mathematics Institute, Cambridge, MA, 2006.
- [CKMW08] F. Cucker, T. Krick, G. Malajovich, and M. Wschebor, *A Numerical Algorithm for Zero Counting I: Complexity and Accuracy*, Journal of Complexity **24** (2008), no. 5-6, 582–605.

- [CKMW09] ———, *A Numerical Algorithm for Zero Counting II: Distance to ill-posedness and Smoothed Analysis*, J. Fixed Point Theory Appl. **6** (2009), no. 2, 285–294.
- [CLO97] D. Cox, J. Little, and D. O’Shea, *Ideals, Varieties, and Algorithms. Undergraduate Texts in Mathematics*, Springer-Verlag, New York, 1997.
- [CPHM01] D. Castro, L.M. Pardo, K. Hägele, and J.E. Morais, *Kronecker’s and newton’s approaches to solving: A first comparison*, J. Complexity **17** (2001), no. 1, 212–303.
- [CR09] D. Costelloe and C. Ryan, *On Improving Generalisation in Genetic Programming*, Genetic Programming, 12th European Conference, EuroGP 2009, Tübingen, Germany, April 15-17, 2009, Proceedings, Lecture Notes in Computer Science, vol. 5481, EuroGP, Springer, 2009, pp. 61–72.
- [Cra85] N.L. Cramer, *A representation for the adaptive generation of simple sequential programs*, Proceedings of the 1st International Conference on Genetic Algorithms (Hillsdale, NJ, USA), L. Erlbaum Associates Inc., 1985, pp. 183–187.
- [Cre93] N.A. Cressie, *Statistics for Spatial Data*, Wiley-Interscience, 1993.
- [CY03] V. Cherkassky and M. Yunkian, *Comparison of Model Selection for Regression*, Neural Computation **15** (2003), 1691–1714.
- [Dan63] G. Dantzig, *Linear Programming and Extensions*, Princeton University Press, 1963.
- [dC56] P.T. de Chardin, *Le Phénomène Humain*, Editions du Seuil, Paris, 1956.
- [DDH06] J. Demmel, I. Dumitriu, and O. Holtz, *Toward accurate polynomial evaluation in rounded arithmetic*, Foundations of computational mathematics, Santander 2005, London Math. Soc. Lecture Note Ser., vol. 331, Cambridge Univ. Press, Cambridge, 2006, pp. 36–105.
- [Ded96] J.P. Dedieu, *Approximate Solutions of Numerical Problems Condition Number Analysis and Condition Number Theorem*, Lectures in Appl. Math. Amer. Math. Soc., Providence **32** (1996), 263–283.
- [Ded06] ———, *Points Fixes, Zéros Et La Méthode de Newton. Collection Mathématiques Et Applications*, Springer-Verlag, 2006.
- [Dég01] J. Dégot, *A Condition Number Theorem for Underdetermined Polynomial Systems*, Math. Comp. **70** (2001), 329–335.
- [Dem88] J.W. Demmel, *The Probability That a Numerical Analysis Problem Is Difficult*, Math. Comp. **50** (1988), 449–480.

- [DL08] C. Durvye and G. Lecerf, *A concise proof of the Kronecker polynomial system solver from scratch*, *Expositiones Mathematicae* **26** (2008), no. 2, 101–139, <http://www.math.uvsq.fr/~lecerf/software/kronecker/index.html>.
- [DM07] J.P. Dedieu and G. Malajovich, *On the Number of Minima of a Random Polynomial*, To appear in *J. of Complexity*, 2007.
- [Dob73] T. Dobzhansky, *Nothing in Biology Makes Sense Except in the Light of Evolution*, *The American Biology Teacher* (1973), 125–129.
- [Ede88] A. Edelman, *Eigenvalues and Condition Numbers of Random Matrices*, *J. Matrix Anal. Appl. SIAM* **9** (1988), 543–560.
- [Ede92] ———, *On the Distribution of a Scaled Condition Number*, *Math. Comp.* **58** (1992), 185–190.
- [EGP00] N. Elezović, C. Giordano, and J. Pečarić, *The best bounds in Gautschi's inequality*, *Math. Inequal. Appl.* (2000), 239–252.
- [EK95] A. Edelman and E. Kostlan, *How many zeros of a random polynomial are real?*, *Bulletin of the American Mathematical Society* **32** (1995), 1–37.
- [ES03] A.E. Eiben and J.E. Smith, *Introduction to Evolutionary Computing*, Springer-Verlag, 2003.
- [EucEC] Euclides, *Elementos* (*Στοιχεῖα*), Desconocido, ~ 300 AEC.
- [EY36] C. Eckart and G. Young, *Approximation of One Matrix by Another of Lower Rank*, *The Psychometrika* (1936), 211–218.
- [Fed69] H. Federer, *Geometric Measure Theory, Die Grundlehren der Mathematischen Wissenschaften, Band 153*, Springer-Verlag, New York, 1969.
- [Fis38] Sir R. Fisher, *Presidential address to the first indian statistical congress*, *First Indian Statistical Congress*, *Sankhya*, 1938.
- [FMM77] G.E. Forsythe, M.A. Malcolm, and C.B. Moler, *Computer methods for mathematical computations*, Prentice-Hall Series in Automatic Computation, Prentice-Hall Inc., Englewood Cliffs, N.J., 1977.
- [For09] L. Fortnow, *The status of the P versus NP problem*, *Commun. ACM* **52** (2009), no. 9, 78–86.
- [FT91] D. Fudenberg and J. Tirole, *Game theory*, The MIT Press, Cambridge, Massachusetts, 1991.
- [Gau09] C.F. Gauß, *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*, Sumtibus F. Perthes et I.H. Besser, 1809.

- [Gau15] ———, *Demonstratio nova altera theorematis omnem functionem algebraicam rationalem integram unius variabilis in factores reales primi vel secundi gradus resolvi posse*, *Societas Regia Scientiarum Gottingensis* **3** (1815), 107–142.
- [GH93] M. Giusti and J. Heintz, *La détermination des points isolés et de la dimension d'une variété algébrique peut se faire en temps polynomial*, *Computational algebraic geometry and commutative algebra (Cortona, 1991)*, *Sympos. Math.*, XXXIV, Cambridge Univ. Press, Cambridge, 1993, pp. 216–256.
- [GHM⁺98] M. Giusti, J. Heintz, J.E. Morais, J. Morgenstern, and L.M. Pardo, *Straight-line programs in geometric elimination theory*, *Journal of Pure and Applied Algebra* **124** (1998), no. 1–3, 101–146.
- [GL46] E. Galois and S. Lie, *Œuvres Mathématiques: Influence de Galois sur le développement des mathématiques*, *Journal des mathématiques pures et appliquées* **XI** (1846), 381–444.
- [GPW03] M. Giusti, L.M. Pardo, and V. Weispfenning, *Algorithms of Commutative Algebra and Algebraic Geometry: Algorithms for Polynomial Ideals and Their Varieties*, *Handbook of Computer Algebra*, Springer Verlag, 2003.
- [Gre81] W. Greub, *Linear Algebra, Graduate Texts in Mathematics*, Springer, 1981.
- [GV04] A. Gabrielov and N. Vorobjov, *Complexity of computations with Pfaffian and Noetherian functions*, *Normal Forms, Bifurcations and Finiteness Problems in Differential Equations*, *NATO Science Series II*, vol. 137, Kluwer, 2004, pp. 211–250.
- [HB09] Ting Hu and W. Banzhaf, *The Role of Population Size in Rate of Evolution in Genetic Programming*, *Genetic Programming, 12th European Conference, EuroGP 2009, Tübingen, Germany, April 15-17, 2009, Proceedings, Lecture Notes in Computer Science*, vol. 5481, EuroGP, Springer, 2009, pp. 85–96.
- [Her26] G. Hermann, *Die Frage der endlich vielen Schritte in der Theorie der Polynomideale*, *Mathematische Annalen* **95** (1926), 736–788, <http://www.springerlink.com/content/UH1638TH08671441>.
- [Hil90] D. Hilbert, *Über der Theorie der Algebraischen Formen*, *Mathematische Annalen* **36** (1890), 473–534.
- [Hil02] ———, *Mathematical problems*, *Bulletin of the American Mathematical Society* **8** (1902), 437–479.
- [Hil30] ———, *Probleme der grundlegung der mathematik*, *Journal Mathematische Annalen* **102** (1930), 1–9.
- [Hir94] M.W. Hirsch, *Differential topology*, *Graduate Texts in Mathematics*, vol. 33, Springer-Verlag, New York, 1994.

- [Hol92] J.H. Holland, *Adaptation in natural and artificial systems*, MIT Press, Cambridge, MA, USA, 1992.
- [How93] R. Howard, *Kinematic Formula in Riemannian Homogeneous Space*, *The Mem. Amer. Math. Soc.* **106** (1993), vi+69.
- [HS80] J. Heintz and C.P. Schnorr, *Testing polynomials which are easy to compute (extended abstract)*, Conference Proceedings of the Twelfth Annual ACM Symposium on Theory of Computing, 28-30 April, Los Angeles, California, USA, ACM, 1980, pp. 262–272.
- [Inf42] L. Infeld, *Quest: The evolution of a scientist*, Readers Union Ltd, 1942.
- [Kan48] L.V. Kantorovič, *On Newton's method for functional equations*, *Doklady Akad. Nauk SSSR (N.S.)* **59** (1948), 1237–1240.
- [KM96] K. Kühnle and E.W. Mayr, *Exponential space computation of gröbner bases*, ISSAC, 1996, pp. 63–71.
- [KM97] M. Karpinski and A. Macintyre, *Polynomial bounds for VC dimension of sigmoidal and general Pffafian neural networks*, *J. Comp. Sys. Sci.* **54** (1997), 169–176.
- [Knu93] D. Knuth, *Computer Literacy Bookshops Interview*, <http://karthikr.wordpress.com/2006/04/06>, 1993.
- [Kos93] E. Kostlan, *On the Distribution of Roots of Random Polynomials*, pp. 419–431, Springer, 1993.
- [Kos02] E. Kostlan, *On the expected number of real roots of a system of random polynomial equations*, *Foundations of Computational Mathematics*, (Hong Kong, 2000), World Sci. Publ., River Edge, NJ, 2002, pp. 149–188.
- [Koz92] J.R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, The MIT Press, 1992.
- [KP94] T. Krick and L.M. Pardo, *Une approche informatique pour l'approximation diophantienne*, *Comptes rendus de l'Académie des sciences. Série 1, Mathématique* (1994), 407–412.
- [KPS01] T. Krick, L.M. Pardo, and M. Sombra, *Sharp estimates for the arithmetic Nullstellensatz*, *Duke Math. J.* **109** (2001), no. 3, 521–598.
- [LA87] P.J.M. Laarhoven and E.H.L. Aarts (eds.), *Simulated annealing: theory and applications*, Kluwer Academic Publishers, Norwell, MA, USA, 1987.
- [Lam09] J-B. Lamarck, *Philosophie zoologique ou exposition des considérations relatives à l'histoire naturelle des animaux*, G. Baillièrre, 1809.
- [Lib46] G. Libri, *Correspondance mathématique*, *Journal des Savants* **Enero** (1846), 50–62.

- [Lug02] G. Lugosi, *Principles of Nonparametric Learning*, ch. Pattern Classification and Learning Theory, pp. 5–62, Springer, 2002.
- [MABC09] J.L. Montaña, C.L. Alonso, C.E. Borges, and J.L. Crespo, *Adaptation, Performance and Vapnik-Chervonenkis Dimension of Straight Line Programs*, Genetic Programming, 12th European Conference, EuroGP 2009, Tübingen, Germany, April 15-17, 2009, Proceedings, Lecture Notes in Computer Science, vol. 5481, EuroGP, Springer, 2009, pp. 315–326.
- [MB09] J.L. Montaña and C.E. Borges, *Lower Bounds for Approximation of Some Classes of Lebesgue Measurable Functions by Sigmoidal Neural Networks*, Bio-Inspired Systems: Computational and Ambient Intelligence, 10th International Work-Conference on Artificial Neural Networks, IWANN 2009, Salamanca, Spain, June 10-12, 2009. Proceedings, Part I, Lecture Notes in Computer Science, vol. 5517, IWANN, Springer, 2009, pp. 1–8.
- [McC04] S. McConnell, *Code Complete: A Practical Handbook of Software Construction*, Microsoft Press, 2004.
- [Meh92] S. Mehrotra, *On the Implementation of a Primal-Dual Interior Point Method*, SIAM Journal on Optimization **2** (1992), no. 4, 575–601.
- [Mel] D.J. Melville, *The famous 'root(2)' tablet from the Yale Babylonian Collection*, <http://it.stlawu.edu/~dmelvill/mesomath/tablets/YBC7289.html>.
- [Mil64] J. Milnor, *On the Betti Numbers of Real Varieties*, Proc. Amer. Math. Soc. **15** (1964), 275–280.
- [MLS94] Z. Michalewicz, T. Logan, and S. Swaminathan, *Evolutionary operators for continuous convex parameter spaces*, Proceedings of the 3rd Annual Conference on Evolutionary Programming, World Scientific, 1994, pp. 84–97.
- [Mor70] G.R. Morrow, *A commentary on the first book of euclid's elements, translated with introduction and notes*, Princeton University Press, 1970.
- [MR04] G. Malajovich and M. Rojas, *High Probability Analysis of the Condition Number of Sparse Polynomial Systems*, Theoret. Comput. Sci. (2004), 524–555.
- [New11] I. Newton, *De analysi per aequationes numero terminorum infinitas*, William Jones, London, 1711.
- [Nor94] P. Nordin, *A compiling genetic programming system that directly manipulates the machine code*, pp. 311–331, MIT Press, Cambridge, MA, USA, 1994.
- [onl] NOVA online, *Solving Fermat: Andrew Wiles*, <http://www.pbs.org/wgbh/nova/proof/wiles.html>.
- [oO95] William of Ockham, *Quaestiones et decisiones in quattuor libros sententiarum petri lombardi*, Ed. Lugd., 1495.

-
- [Ost54] A. Ostrowski, *On two problems in abstract algebra connected with Horner's rule*, Studies in mathematics and mechanics presented to Richard von Mises, Academic Press Inc., New York, 1954, pp. 40–48.
- [OZ05] S. Olariu and A.Y. Zomaya, *Handbook of Bioinspired Algorithms and Applications*, Chapman & Hall/Crc Computer & Information Science, Chapman & Hall/CRC, 2005.
- [Pan97] V.Y. Pan, *Solving a Polynomial Equation: Some History and Recent Progress*, SIAM Review **39** (1997), no. 2, 187–220.
- [Pap94] C.H. Papadimitriou, *Computational complexity*, Addison-Wesley Publishing Company, Reading, MA, 1994.
- [Par95] L.M. Pardo, *How lower and upper complexity bounds meet in elimination theory*, Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, 11th International Symposium, AAEEC-11, Paris, France, July 17–22, 1995, Proceedings, Lecture Notes in Computer Science, vol. 948, 1995, pp. 33–69.
- [PFTV92] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, *Numerical recipes in fortran 77: The art of scientific computing*, ch. 16.7, pp. 740–744, Cambridge University Press, 1992.
- [PLW05] T. Plewa, T.J. Linde, and V.G. Weirs (eds.), *Adaptive mesh refinement, theory and applications: proceedings of the Chicago Workshop on Adaptive Mesh Refinement Methods, Sept. 3–5, 2003*, Lecture Notes in Computational Science and Engineering, vol. 41, Springer, 2005.
- [PMCC09] R. Poli, N.F. McPhee, L. Citi, and E.F. Crane, *Memory with memory in tree-based genetic programming.*, Genetic Programming, 12th European Conference, EuroGP 2009, Tübingen, Germany, April 15–17, 2009, Proceedings, Lecture Notes in Computer Science, vol. 5481, EuroGP, Springer, 2009, pp. 25–36.
- [Pol08] R. Poli, *Analysis of the publications on the applications of particle swarm optimisation*, J. Artif. Evol. App. **2008** (2008), 1–10.
- [Pow07] W.B. Powell, *Approximate dynamic programming*, Wiley Series in Probability and Statistics, Wiley-Interscience [John Wiley & Sons], Hoboken, NJ, 2007, Solving the curses of dimensionality.
- [Pri06] U. Prior, *Erwartete Anzahl Reeller Nullstellen von Zufälligen Polynomen*, Master's thesis, Institut für Mathematik der Universität Paderborn, 2006.
- [Pro50] Proclo, *Comentarios sobre el primer libro de Los Elementos (Στοιχεῖα)*, Desconocido, ~450.
- [PSL05] K. Price, R.M. Storn, and J.A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, 1 ed., Springer, 2005.
- [Ras06] C.E. Rasmussen, *Gaussian processes for machine learning*, MIT Press, 2006.

- [Ren94] J. Renegar, *Is It Possible to Know a Problem Instance Is Ill-posed? Some Foundations for a General Theory of Condition Numbers*, J. Complexity **10** (1994), 1–56.
- [RMH80] G.C. Rota, N. Metropolis, and J. Howlett, *A History of computing in the twentieth century: a collection of essays*, Academic Press, New York, 1980 (English).
- [Run01] C. Runge, *Über empirische Funktionen und die Interpolation zwischen äquidistanten ordinaten*, Zeitschrift für Mathematik und Physik **46** (1901), 224–243.
- [Sch78] G. Schwarz, *Estimating the dimension of a model*, The Annals of Statistics **6** (1978), no. 2, 461–464.
- [Sch81] H-P. Schwefel, *Numerical optimization of computer models*, John Wiley & Sons, Inc., New York, NY, USA, 1981.
- [Sim37] T. Simpson, *A New Treatise of Fluxions*, Thos. Gardner, et al., London, 1737.
- [Sma81] S. Smale, *The fundamental theorem of algebra and complexity theory*, Bull. Amer. Math. Soc. (N.S.) **4** (1981), no. 1, 1–36.
- [Sma85] S. Smale, *On the Efficiency of Algorithms of Analysis*, Bull. Amer. Math. Soc. (1985), 87–121.
- [Sma00] ———, *Mathematical Problems for the Next Century*, Mathematics: Frontiers and Perspectives, Amer. Math. Soc., Providence, 2000, pp. 271–294.
- [SS93a] M. Shub and S. Smale, *Complexity of Bézout’s Theorem I: Geometric Aspects*, J. of the Amer. Math. Soc. **6** (1993), 459–501.
- [SS93b] ———, *Complexity of Bézout’s Theorem II: Volumes and Probabilities*, Progr. Math. **109** (1993), 267–285.
- [SS93c] ———, *Complexity of Bézout’s Theorem III: Condition Number and Packing*, J. Complexity **9** (1993), 4–14.
- [SS94] ———, *Complexity of Bézout’s Theorem V: Polynomial Time*, Theoret. Comput. Sci. **133** (1994), 141–164.
- [SS96] ———, *Complexity of Bézout’s Theorem IV: Probability of Success and Extensions*, J. Numer. Anal. SIAM **33** (1996), 128–148.
- [Ste08] Ian Stewart, *Belleza y verdad: Una historia de la simetría*, Crítica, 2008.
- [Str] B. Stroustrup, *Bjarne stroustrup’s faq*, http://www2.research.att.com/~bs/bs_faq.html.

-
- [Str90] V. Strassen, *Algebraic Complexity Theory*, Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity, The MIT Press, 1990, pp. 633–672.
- [Tea] R Development Core Team, *The R Manuals*, <http://cran.r-project.org/manuals.html>.
- [TGBS05] O. Teytaud, S. Gelly, N. Bredeche, and M.A. Schoenauer, *Statistical Learning Theory Approach of Bloat*, Proceedings of the 2005 conference on Genetic and Evolutionary Computation, 2005, pp. 1784–1785.
- [Tur37a] A. Turing, *Computability and λ -definability*, Journal of Symbolic Logic (1937), 153–163.
- [Tur37b] ———, *On computable numbers, with an application to the entscheidungsproblem*, Proceedings of the London Mathematical Society (1937), 230–65.
- [Tur48] ———, *Rounding-off Errors in Matrix Processes*, Quart. J. Mech. Appl. Math. (1948), 287–308.
- [Tur69] ———, *Intelligent machinery*, Machine Intelligence **5** (1969), 3–26.
- [Vap98] V. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, 1998.
- [VC71] V. Vapnik and A. Chervonenkis, *On the uniform convergence of relative frequencies of events to their probabilities*, Theory of Probability and its Applications **16** (1971), 264–280.
- [VC74] ———, *Ordered risk minimization*, Automation and Remote Control **34** (1974), 1226–1235.
- [vN45] J. von Neumann, *First Draft of a Report on the EDVAC*, Tech. report, United States Army Ordnance Department, 1945.
- [vNG47] J. von Neumann and H. Goldstine, *Numerical Inverting of Matrices of High Order*, Bull. Amer. Math. Soc. (1947), 1021–1099.
- [vzG88] J. von zur Gathen, *Algebraic Complexity Theory*, pp. 317–347, Annual Reviews Inc., Palo Alto, CA, USA, 1988.
- [vzGG03] J. von zur Gathen and J. Gerhard, *Modern computer algebra*, second ed., Cambridge University Press, Cambridge, 2003.
- [Wik10] Wikipedia, *Biologically inspired computing — wikipedia, the free encyclopedia*, 2010, [Online; accessed 27-October-2010].
- [Wil65] J.H. Wilkinson, *The algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.

BIBLIOGRAFÍA

- [WLJ01] R.P. Wiegand, W.C. Liles, and K.A. De Jong, *An empirical analysis of collaboration methods in cooperative coevolutionary algorithms*, Proceedings from the Genetic and Evolutionary Computation Conference, Morgan Kaufmann, 2001, pp. 1235–1242.
- [WM97] D.H. Wolpert and W.G. Macready, *No free lunch theorems for optimization*, IEEE Transactions on Evolutionary Computation **1** (1997), no. 1, 67–82.
- [Zuk84] G. Zukav, *The Dancing Wu Li Masters: An Overview of the New Physics*, Bantam, 1984.

Índice de pseudocódigos

1	Pseudocódigo de un método evolutivo.	xxx1
2	Pseudocódigo de un algoritmo de escalada.	xxx1
3	Pseudocódigo de un algoritmo genético.	xxxii
4	Pseudocódigo de un algoritmo de enjambre de partículas.	xxxiii
8.1	Pseudocódigo para el cálculo de la aplicación semántica.	74
10.1	Pseudocódigo del proceso co-evolutivo.	106
12.1	Pseudocódigo para el algoritmo de búsqueda exhaustiva.	121
12.1	Pseudocódigo para el algoritmo de búsqueda exhaustiva con refinamiento del retículo.	122

Índice de figuras

1	Sucesión de Newton.	xxiv
2	Ejemplo de problema de regresión simbólica.	xxxv
3	Representación de f como árbol.	xxxvi
4	Árboles hijos mediante el cruce de arboles	xli
8.1	Grafo dirigido que representa un slp	70
8.2	Bosque asociado al slp Γ del Ejemplo 8.1.10.	72
8.3	Slp's padres.	77
8.4	Slp's hijos mediante el cruce en un punto.	77
8.5	Slp's hijos mediante el cruce en dos puntos.	77
8.6	Slp's hijos mediante el cruce uniforme.	78
8.7	Slp's hijos mediante el cruce slp	79
8.8	<i>Diagramas de Cajas y Bigotes</i> : cruces para slp	81
8.9	<i>Evolución de la Calificación</i> : cruces para slp	83
8.10	<i>Diagramas de Cajas y Bigotes</i> : slp frente a <i>arbol</i>	84
9.1	<i>Diagramas de Cajas y Bigotes</i> : métodos de calificación con ruido.	98
9.2	<i>Función de densidad empírica</i> : métodos de calificación con ruido.	100
9.3	<i>Función de densidad empírica</i> : métodos de calificación con ruido.	101
10.1	<i>Diagramas de Cajas y Bigotes</i> : calificación de constantes.	108
10.2	<i>Diagramas de Cajas y Bigotes</i> : algoritmos de co-evolución.	110
11.1	Slp's hijos mediante el cruce slp extendido.	115
12.1	Ejemplo de sistema con mal comportamiento.	118

Índice de cuadros

8.1	<i>Parámetros específicos: cruces para slp.</i>	81
8.2	<i>Tasas de error y de solución: cruces para slp.</i>	81
8.3	<i>Resumen estadístico I: cruces para slp.</i>	82
8.4	<i>Resumen estadístico II: cruces para slp.</i>	82
8.5	<i>Contraste de Hipótesis: cruces para slp.</i>	82
8.6	<i>Parámetros específicos: slp frente a arbol.</i>	83
8.7	<i>Tasas de error y de solución: slp frente a arbol.</i>	84
8.8	<i>Resumen estadístico I: slp frente a arbol.</i>	84
8.9	<i>Resumen estadístico II: slp frente a arbol.</i>	84
8.10	<i>Contraste de Hipótesis: slp frente a arbol.</i>	85
9.1	<i>Parámetros específicos: métodos de calificación con ruido.</i>	97
9.2	<i>Tasas de error y de solución: métodos de calificación con ruido.</i>	98
9.3	<i>Resumen estadístico I: métodos de calificación con ruido.</i>	99
9.4	<i>Resumen estadístico II: métodos de calificación con ruido.</i>	99
9.5	<i>Contraste de Hipótesis: métodos de calificación con ruido.</i>	99
10.1	<i>Parámetros específicos: calificación de constantes.</i>	107
10.2	<i>Tasas de error y de solución: calificación de constantes.</i>	107
10.3	<i>Resumen estadístico I: calificación de constantes.</i>	108
10.4	<i>Resumen estadístico II: calificación de constantes.</i>	108
10.5	<i>Contraste de Hipótesis: calificación de constantes.</i>	109
10.6	<i>Parámetros específicos: algoritmos de co-evolución.</i>	109
10.7	<i>Tasas de error y de solución: algoritmos de co-evolución.</i>	109
10.8	<i>Resumen estadístico I: algoritmos de co-evolución.</i>	110
10.9	<i>Resumen estadístico II: algoritmos de co-evolución.</i>	110
10.10	<i>Contraste de Hipótesis: algoritmos de co-evolución.</i>	111
12.1	<i>Parámetros algoritmo evolutivo genético para la resolución del problema XVII de Smale.</i>	125

ÍNDICE DE CUADROS

12.2	Parámetros algoritmo evolutivo diferencial para la resolución del problema XVII de Smale.	126
12.3	Parámetros algoritmo evolutivo basado en enfriamiento simulado para la resolución del problema XVII de Smale.	126
12.4	Parámetros algoritmo evolutivo basado en enjambres de partículas para la resolución del problema XVII de Smale.	126
12.5	Parámetros descriptivos de los sistemas de ecuaciones.	127
12.6	Resultados algoritmos evolutivos genéticos.	127
12.7	Resultados algoritmos evolutivos diferenciales.	128
12.8	Resultados algoritmos de enfriamiento simulado.	128
12.9	Resultados algoritmos de enjambre de partículas.	128
B.1	Parámetros comunes.	140
B.2	Parámetros comunes muestras polinómicas.	140
B.3	Parámetros comunes muestras $slp's$	140

Glosario

A

acción 54–56

algoritmo bioinspirado xxx, xxxi, xxxv

algoritmo evolutivo xxx, xxxiii, xxxiv, xxxix, xl, 103, 104, 106, 107, 113, 115, 117, 122–125, 137, 140

enfriamiento simulado xl, 124

enjambre de partículas xl, 124, 129

evolución diferencial xl, 123

programación genética xxxii–xxxv, xxxvii, 67, 68, 74–76, 80, 81, 83, 88, 94, 95, 97, 104, 106, 107, 111, 113, 116, 137, 138, 140

programación genética lineal 67

aplicación semántica $(\Phi(\Gamma))$ Aplicación que resulta de evaluar el código efectivo de un slp Γ . 69, 70, 72–74, 76, 88, 90, 91, 138

aplicación tangente $(T_x f)$ Aplicación tangente de f en el punto x . 6

árbol Grafo en el que dos vértices están conectados por exactamente un camino. xxxvi, xxxvii, 67, 68, 70–72, 74, 75, 80, 83–85, 95, 113, 137, 139

aridad Número de parámetros de un función. 68, 69

B

base Conjunto de vectores generadores de un espacio vectorial y linealmente independientes.

base ortonormal Base de un espacio vectorial tal que sus elementos son ortogonales dos a dos y de norma unitaria respecto a una norma. 10, 28

base de Gröbner 4

base monomial Base del conjunto de monomios de un polinomio. 4

bola cerrada $(\overline{B}(c, r))$ Bola cerrada de centro c y radio r . 31, 33

bosque Conjunto de árboles. 71, 72

C

código efectivo Instrucciones que son necesarias a la hora de evaluar la aplicación semántica de un slp Γ . 73, 78, 114, 115

código no efectivo Instrucciones que no son necesarias a la hora de evaluar la aplicación semántica de un slp Γ . 72, 73, 75, 114

cadena de Pfaffian Funciones especiales que son solución de ecuaciones diferenciales polinomiales. 87, 89–92, 94

carta afín ($\varphi_f(g)$) Carta afín.. 6, 25
clasificación 95
codimensión ($\text{codim}(V)$) Codimensión de una variedad diferenciable V . 7, 11, 64
codificación densa Codiciación de un polinomio como el vector de sus coeficientes. 5
coeficiente multinomial $\binom{l}{\mu} := \frac{l!}{\mu_0! \cdots \mu_n!}$. 5
complemento ortogonal (x^\dagger) Complemento ortogonal del punto x . 6, 15
componente conexa xxv, xxvi, 8, 45–48, 50, 52–56, 59, 63, 64, 89, 90, 94
concepto xxxiv
condicionamiento
 condicionamiento lineal ($\kappa(A)$) Condicionamiento lineal de la matriz A . xx, 12–16
 condicionamiento μ medio ($\mu_{norm}^{av}(f)$) μ normalizado medio del sistema f . 42
 condicionamiento μ_{norm} ($\mu_{norm}(f, x)$) μ normalizado del sistema f en el punto x . xx, xxi, xxiii, 14–16, 37, 42, 61, 63
 condicionamiento μ peor ($\mu_{worst}(f)$) Peor μ normalizado del sistema f . xxi, xxii
conexo 49, 50, 53
conjunto cuestor (\mathcal{G}) xxv, 64
cono (\tilde{A}) Cono del subconjunto A . 10
convexo 46
curva isobárica Curva sobre la cual una función es constante. 59

D

número de Bézout (\mathcal{D}) Número de soluciones en la clausura algebraica de un sistema de ecuaciones polinomiales. Su valor es $\prod_{i=1}^n d_i$. xxvi, xxvii, xxxix, 12, 33, 120
deformación homotópica xvii, xxiii, 45, 59, 64
denso 25
descenso de gradiente xxxviii, 115
diferencial ($\mathcal{D}_x f$) Diferencial de la función f en el punto x . xix
dimensión ($\text{dim}(V)$) Dimensión de una variedad diferenciable V . 7, 11, 31

E

error empírico Error cuadrático medio sobre la muestra de aprendizaje. 76, 95, 96, 115
error teórico Error que tiene en cuenta tanto el error empírico como una penalización que tenga en cuenta la complejidad del modelo. 95, 96
espacio de Banach Espacio de Banach. xviii
espacio de medida 12
espacio tangente ($T_x V$) Espacio tangente de la variedad diferenciable V en el punto x . 6
espacio topológico 45
espacio vectorial
 esfera ($\mathbb{S}_d^r(\mathbb{K})$) Esfera de dimensión d y radio r sobre el cuerpo \mathbb{K} . 48
 matriz ($\mathcal{M}_{m \times n}(\mathbb{K})$) Espacio de las matrices de dimensiones $n \times m$ sobre el cuerpo \mathbb{K} . 15
 matriz cuadrada ($\mathcal{M}_n(\mathbb{K})$) Espacio de las matrices cuadradas de dimensión n sobre el cuerpo \mathbb{K} . 13

matriz ortogonal ($\mathcal{O}(d, \mathbb{K})$) Espacio de las matrices ortogonales de dimensión d sobre el cuerpo \mathbb{K} . 7, 11

espacio proyectivo ($\mathbb{P}_d(\mathbb{K})$) Espacio proyectivo de dimensión d sobre el cuerpo \mathbb{K} . 4, 6, 48

espacio proyectivo 9, 10

espacio proyectivo de sistemas homogéneos ($\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{K}})$) Espacio proyectivo de sistemas de ecuaciones polinomiales multivariadas homogéneas con lista de grados (d) y con coeficientes en el cuerpo \mathbb{K} . 5, 6

polinomios ($\mathbb{P}_d^{\mathbb{K}}$) Espacio vectorial de los polinomios multivariados de grado menor que d y con coeficientes en el cuerpo \mathbb{K} . 3

polinomios homogéneos ($\mathbb{H}_d^{\mathbb{K}}$) Espacio vectorial de polinomios homogéneos multivariados de grado d y con coeficientes en el cuerpo \mathbb{K} . 3, 4

sistema de ecuaciones polinomiales homogéneas multivariadas ($\mathcal{H}_{(d)}^{\mathbb{K}}$) Espacio vectorial de los sistemas de ecuaciones polinomiales homogéneas multivariadas con lista de grados (d) y con coeficientes en el cuerpo \mathbb{K} . xxi, xxv, xxvi, 3, 14, 50, 63, 64

sistema de ecuaciones polinomiales multivariadas ($\mathcal{P}_{(d)}^{\mathbb{K}}$) Espacio vectorial de los sistemas de ecuaciones polinomiales multivariadas con lista de grados (d) y con coeficientes en el cuerpo \mathbb{K} . xvii, xx, xxi, xxv–xxvii, xxxix, 3, 25, 34, 45

sistemas de ecuaciones lineales Espacio vectorial de los sistemas de ecuaciones lineales con coeficientes en el cuerpo \mathbb{K} . xx, 46

esperanza ($E_V[f]$) Esperanza de la función f en el Espacio V . xxvi, 9

esquema xxxiii

F

fibra ($V_x\mathbb{R}$) Fibra del punto x por la proyección π_2 . 7, 8

fórmula de la coárea 10, 24, 28–30, 32

función (\mathcal{F}) Conjunto de operadores que pueden aparecer en un árbol o *slp*. xxxvi, xxxvii, 68, 69, 74, 75, 79, 87, 90, 94, 97, 115, 125, 138, 140

función característica (χ_A) Función característica del subconjunto A . 9, 10, 42

función de pertenencia (Φ) Función que comprueba si una muestra pertenece al concepto. 88, 89

G

grado ($\deg(p)$) Grado del polinomio p . xxii, 3, 8, 9

grupo

grupo ortogonal Grupo de las matrices ortogonales. 5–8, 15, 22, 24

grupo ortogonal especial ($S^+\mathcal{O}(d, \mathbb{K})$) Grupo ortogonal especial de dimensión d sobre el cuerpo \mathbb{K} . 9, 46, 54, 59

H

heurístico xxx–xxxiv, 125, 129

hiperplano 25

I

invariante Una función f es *invariante* en el conjunto C por la acción del grupo G si f tiene el mismo valor en las órbitas de G . 6, 22, 24, 55, 56

isometría Aplicación que conserva las propiedades métricas. 6–8, 10, 15–18, 26

J

jacobiano extendido orientado ($\Delta_x f$) Jacobiano extendido orientado de la función f en el punto x . xxv, 52

jacobiano normal ($\text{NJ}_x f$) Jacobiano normal de la función f en el punto x . 10, 11, 19, 22, 26, 29

M

máquina de aprendizaje automático Algoritmo que automáticamente aprende a extraer conocimientos en distintos contextos. 88, 95, 103, 113

matriz antisimétrica 59

matriz de Kostlan (Δ_d) 5

matriz Jacobiana ($D_x(f)$) Matriz Jacobiana de la función f en el punto x . xviii, 6, 119

medida de Hausdorff ($\nu_d[A]$) Medida de Hausdorff del conjunto A . xxi, 9

medida nula Conjunto que tiene dimensión inferior a la variedad que lo contiene. 8

método de Newton

cero aproximado Punto $x \in \mathbb{P}_n(\mathbb{R})$ tal que la iteración de Newton asociada converge cuadráticamente hacia una raíz. xvii, xix, xx, xxii, xxiii, xxvi, xxxix, xl, 118–123

iteración de Newton Sucesión de puntos que surge de iterar el operador de Newton sobre un punto. xviii–xx, xxiii, xxxix, 119, 120

operador de Newton ($N_f^k(x)$) Operador de Newton de la aplicación f sobre el punto x k veces. xviii, xxvi, 117, 120

radio de convergencia Radio máximo de bola que contiene ceros aproximados. xix, xxii

sucesión de Newton Sucesión de puntos que surge de seguir una curva con el operador de Newton. xxiii–xxv, 63, 64

modelo xxxiv, xxxv

muestra aleatoria simple Muestra i.i.d. de elementos en una población. 137, 138

muestra de aprendizaje Muestra sobre la que se entrenará una máquina de aprendizaje automático. 94–96

N

número de Betti ($\beta_0(V)$) Número de componentes conexas por caminos del conjunto V . 45

navaja de Ockham Principio filosófico según el cual han de preferirse las teorías más simples a las más complejas. 97

norma

norma ($\|v\|$) Norma del vector v . 16

norma de Frobenius ($\|A\|_F$) Norma de Frobenius de la matriz A . 13–16, 48

O

operador genético

- calificación** xxx–xxxiii, 75, 76, 81, 83, 88, 94, 97–100, 103–105, 107, 109, 113–115, 117, 118, 122–124, 127–129, 142
- α^* Calificación de constantes que toma como valor el resultado de $\alpha^*(f, x)$. 128
- α_c^* Calificación de constantes que toma como valor el resultado de $\alpha^*(f, x)$ corregido mediante la iteración del operador de Newton. 128, 129
- élite** Calificación de constantes que toma el valor de la calificación de la élite de la población de *slp*'s al sustituir los parámetros. 105, 107, 109
- media** Calificación de constantes que toma el valor medio entre la calificación de la élite de la población de *slp*'s y otro *slp* elegido al azar al sustituir los parámetros. 105, 107, 109
- mínimo** Calificación de constantes que toma el valor mínimo entre la calificación de la élite de la población de *slp*'s y otro *slp* elegido al azar al sustituir los parámetros. 105, 107, 109
- normal** Calificación de constantes que toma el mínimo de la calificación de la población de *slp*'s al sustituir los parámetros. 104, 107, 109
- cruce** xxxii, xxxvii, 78, 80, 81, 83, 97, 105, 107, 109, 114, 123–126
- árbol** Cruce que traslada subexpresiones entre dos árboles manteniendo la longitud constante. xxxvii, 83
- aritmético** Cruce que realiza una combinación convexa entre las constantes. 105, 107, 109, 123, 125, 126
- diferencial** Cruce que realiza una combinación lineal entre tres individuos. 123, 126
- en dos puntos** Cruce que intercambia las posiciones entre dos puntos de un *slp* al otro (y viceversa). 76, 77, 79–81, 83
- en un punto** Cruce que intercambia las posiciones desde un punto al final de un *slp* al otro (y viceversa). 76, 77, 79–81, 83
- enjambre** Cruce que realiza una combinación lineal entre el individuo actual, el mejor individuo de la población y el mejor individuo por el que ha pasado. 124, 126
- slp** Cruce que traslada subexpresiones entre dos *slp*'s manteniendo la longitud constante. 78–83, 97, 107, 109, 114
- slp extendido** Cruce que traslada subexpresiones entre dos *slp*'s sin mantener la longitud constante. 114, 115
- uniforme** Cruce que intercambia de forma aleatoria los elementos de la misma posición entre dos *slp*'s. 76, 77, 79–81, 83
- enfriamiento simulado** Si el nuevo individuo es mejor lo selecciona. Si es peor, lo selecciona con cierta probabilidad. 124, 126
- migración** 116
- mutación** xxxii, xxxvii, 79–81, 83, 97, 105, 107, 109, 123, 125, 126
- árbol** Mutación que sustituye un función de un árbol por otro. xxxvii, 83
- no uniforme** Mutación de constantes que explora el espacio global en las primeras generaciones mientras que en las últimas lo hace localmente. 105, 107, 109, 123, 125, 126
- slp** Mutación que sustituye una subexpresión de un *slp* por otra. 79, 81, 83, 97, 107, 109
- por ranqueo** Selecciona a los mejores individuos, todos ellos distintos, entre padres e hijos. 125, 126, 140
- reemplazo** 79, 80, 105, 124–126, 140

reproducción xxxi, xxxvii, 79

selección xxxi, xxxvii, 80, 105, 125, 126, 140

q -torneo Escoge los mejores individuos de entre un grupo de q individuos escogidos al azar. 80, 105

ruleta Escoge los individuos para realizar un operador genético con probabilidad proporcional a su calificación. 80

simple Escoge los individuos secuencialmente. 80, 125, 126, 140

orden monomial (\leq_l) Ordenación del conjunto de monomios de un polinomio. 4, 5

P

probabilidad ($\text{Prob}_V[C]$) Probabilidad suceso C en el espacio V . 9

producto escalar

producto escalar de Bombieri ($\langle x, y \rangle_\Delta$) Producto escalar de Bombieri entre los vectores \mathbf{x} e \mathbf{y} . 4–6, 14–16, 22, 23, 52

producto escalar euclídeo ($\langle x, y \rangle$) Producto escalar euclídeo entre los vectores \mathbf{x} e \mathbf{y} . 5, 9

proyección canónica (π) Proyección canónica. xviii, 7, 10

pseudo-inversa de Moore-Penrose (M^\dagger) Pseudo-inversa de Moore-Penrose de la matriz M . xviii, 12–14

punto regular Puntos regulares de una submersión. 8, 10

punto singular Puntos singulares de una submersión. 7, 8, 15

R

rango ($\text{rank}(M)$) Rango de una matriz M . 8

regresión simbólica xxxiv, xxxvii, 68, 74, 80, 95, 97, 113, 124, 138

regresor xxxiv

resultante x

retículo 121, 122, 129

S

sistema factible ($\mathcal{R}_{(d)}$) Conjunto de sistemas de ecuaciones polinomiales de grados (d) sin raíces singulares. 8, 63

sistemas numéricos

número complejo (\mathbb{C}) Conjunto de los números complejos. ix, xvii, 31, 131–133

número natural (\mathbb{N}) Conjunto de los números naturales. xxiii, xxvii, 3–5, 15, 24–26, 28, 41, 46, 51, 64, 114, 131, 132, 134

número real (\mathbb{R}) Conjunto de los números reales. xxi–xxiii, xxvi, xxvii, xxxix, 3, 4, 8, 9, 28, 31, 33, 41, 42, 88, 92, 105, 123, 124, 134

número entero (\mathbb{Z}) Conjunto de los números enteros. x , 132

submersión 7, 8, 10

suma ortogonal (\oplus) Suma ortogonal de dos subespacios vectoriales. 15

T

tasa de éxito (t_s) Porcentaje de ejecuciones de en las que la calificación de validación del modelo es inferior al 10% del rango de la muestra. 81, 83, 97, 107, 109, 141

tasa de error (t_e) Porcentaje de ejecuciones de en las que la calificación de validación del modelo es superior al 150% del rango intercuartílico. 81, 83, 97, 107, 109, 141

terminal (\mathcal{T}) Conjunto de terminales que pueden un árbol o *slp*. xxxvi, xxxvii, 68, 69, 71, 74, 75, 87, 90, 97, 103, 114, 115, 125, 138, 140

topología de Zariski 11, 12

V

valor crítico Valores críticos de una submersión. ix, 7, 8

valor regular Valores regulares de una submersión. 90

variable aleatoria 12, 41

variable no terminal Instrucciones que constituyen un *slp* Γ . 69–75, 78, 79, 91–93, 114, 115, 140

variedad

subvariedad de Riemann 9

variedad algebraica afín ($V_{\mathbb{A}}^{\mathbb{K}}(f)$) variedad algebraica afín de los ceros comunes, pertenecientes al cuerpo \mathbb{K} , del sistema de polinomios $f \in \mathcal{P}_{(d)}^{\mathbb{K}}$. 3

variedad algebraica lisa Variedad algebraica que no contiene punto singular. 11

variedad de incidencia ($V^{\mathbb{K}}$) Variedad de incidencia sobre el cuerpo \mathbb{K} . xxii, xxv, 4, 7, 14, 15, 17, 45, 54, 59

variedad de incidencia mixta ($V^{\mathbb{R}, \mathbb{C}}$) Variedad de incidencia de las soluciones complejas de sistemas pertenecientes a $\mathbb{P}(\mathcal{H}_{(d)}^{\mathbb{R}})$. 4

variedad de Riemann 6, 9, 10

variedad discriminante ($\Sigma^{\mathbb{K}}$) Variedad de los sistemas de ecuaciones que poseen alguna raíz singular en el cuerpo \mathbb{K} . xxv, 8, 45

variedad multi-homogénea 4, 7

variedad algebraica proyectiva ($V_{\mathbb{P}}^{\mathbb{K}}(f)$) Variedad algebraica proyectiva de los ceros comunes, pertenecientes al cuerpo \mathbb{K} , del sistema de polinomios $f \in \mathcal{H}_{(d)}^{\mathbb{K}}$. 4, 8

volumen

forma de volumen ($dv_n^{\mathbb{K}}$) Forma de volumen de dimensión n sobre el cuerpo \mathbb{K} . 9

Siglas

C

criterio de información Bayesiano (*BIC*) 94, 96, 97, 99, 113

criterio de información de Akaike (*AIC*) 94, 96, 97, 99, 113

E

error cuadrático medio (*MSE*) Error cuadrático medio. xxxv, 74, 76, 81, 83, 95, 97–100, 107, 109, 139

F

función de distribución empírica (*ecdf*) Es una función de distribución de probabilidad que acumula una probabilidad constante (e igual a $1/n$) para cada uno de los n valores de la muestra. 100, 142

I

independientes e igualmente distribuidas (i.i.d.) Variables aleatorias independientes e igualmente distribuidas. 74, 94

M

minimización del riesgo estructural (*SRM*) 88, 94–99

S

stright-line program (*slp*) Grafo dirigido acíclico. xiii, xiv, xxxvii, xxxviii, 67–85, 87, 88, 90, 91, 94, 95, 97–99, 103–107, 109, 111, 113–115, 125, 137–140, 142

V

dimensión de Vapnik-Chervonenkis (VC_{dim}) Dimensión de Vapnik-Chervonenkis. xxxviii, 87–89, 94, 96, 97, 103, 113