EST. 2021 EMC
EDITORIAL MAR CARIBE

Matemáticas para el aprendizaje de máquina

Ángel Amado Romero Cahuana
Rosa Luz Medina Aguilar
Edinson Raúl Montoro Alegre
Domingo Guzmán Chumpitaz Ramos
Juan Honorato Luna Valdez
Olmedo Pizango Isuiza

ISBN: 978-9915-698-74-8
9 789915 698748

Matemáticas para el aprendizaje de máquina

Romero Cahuana, Ángel Amado; Medina Aguilar, Rosa Luz; Montoro Alegre, Edinson Raúl; Chumpitaz Ramos, Domingo Guzmán; Luna Valdez, Juan Honorato; Pizango Isuiza, Olmedo

© *Romero Cahuana, Ángel Amado; Medina Aguilar, Rosa Luz; Montoro Alegre, Edinson Raúl; Chumpitaz Ramos, Domingo Guzmán; Luna Valdez, Juan Honorato; Pizango Isuiza, Olmedo, 2026*

Primera edición (1.ª ed.): febrero, 2026

Editado por:

Editorial Mar Caribe ®

www.editorialmarcaribe.es

Av. Gral. Flores 547, 70000 Col. del Sacramento, Departamento de Colonia, Uruguay.

Diseño de carátula e ilustraciones: *Luisa Fernanda Lugo Rojas*

Libro electrónico disponible en:

https://editorialmarcaribe.es/ark:/10951/isbn.9789915698748

Formato: Electrónico

ISBN: 978-9915-698-74-8

ARK: ark:/10951/isbn.9789915698748

Editorial Mar Caribe (OASPA): Como miembro de la Open Access Scholarly Publishing Association, apoyamos el acceso abierto de acuerdo con el código de conducta, la transparencia y las mejores prácticas de OASPA para la publicación de libros académicos y de investigación. Estamos comprometidos con los más altos estándares editoriales en ética y deontología, bajo la premisa de «Ciencia Abierta en América Latina y el Caribe»

Editorial Mar Caribe, firmante Nº 795 de 12.08.2024 de la Declaración de Berlín

"... Nos sentimos obligados a abordar los retos de Internet como un medio funcional emergente para la distribución del conocimiento. Obviamente, estos avances pueden modificar significativamente la naturaleza de la publicación científica, así como el actual sistema de garantía de calidad...." (Max Planck Society, ed. 2003, pp. 152-153).

CC BY-NC 4.0

Los autores pueden autorizar al público en general a reutilizar sus obras únicamente con fines no lucrativos, los lectores pueden utilizar una obra para generar otra, siempre que se dé crédito a la investigación, y conceden al editor el derecho a publicar primero su ensayo bajo los términos de la licencia CC BY-NC 4.0.

Editorial Mar Caribe se adhiere a la *"Recomendación relativa a la preservación del patrimonio documental, comprendido el patrimonio digital, y el acceso al mismo"* de la UNESCO y a la Norma Internacional de referencia para un sistema abierto de información archivística (OAIS-ISO 14721). Este libro está preservado digitalmente por ARAMEO.NET

Editorial Mar Caribe

Matemáticas para el aprendizaje de máquina

Colonia, Uruguay

2026

Matemáticas para el aprendizaje de máquina

Índice

Introducción

El ascenso y la consolidación del aprendizaje automático (Machine Learning) en las primeras décadas del siglo XXI han revolucionado la estructura de la sociedad moderna, igualando en impacto hitos históricos como la invención del motor de combustión interna o el avance en las comunicaciones inalámbricas. Actualmente, los algoritmos de aprendizaje automático están presentes en muchas áreas, desde la lectura de matrículas en cámaras de tráfico hasta la optimización de cadenas de suministro globales y la personalización de contenidos en redes sociales.

Sin embargo, bajo la superficie de estas aplicaciones tecnológicas se encuentra un lenguaje fundamental y riguroso: el de las matemáticas. La relación entre las matemáticas y el aprendizaje automático no es meramente instrumental; es ontológica. Las matemáticas no son solo una herramienta para construir modelos, sino el lenguaje mismo que permite a los sistemas artificiales aprender, adaptarse y tomar decisiones en entornos de incertidumbre.

Para comprender la magnitud de esta disciplina, se puede emplear la analogía del Burj Khalifa: así como el edificio más alto del mundo requiere cimientos extraordinariamente profundos y sólidos para mantenerse estable, la inteligencia artificial requiere una base matemática sólida para crecer y evolucionar de manera segura y eficiente. Sin estos cimientos, cualquier desarrollo tecnológico en el área es propenso a la inestabilidad, la falta de precisión y la incapacidad de generalizar.

Esta obra de investigación se propone no solo como un compendio técnico, sino también como una respuesta a la creciente necesidad de

profesionales que no solo operen software, sino que comprendan la mecánica interna de los algoritmos que están redefiniendo la civilización.

Históricamente, el camino hacia el aprendizaje automático ha sido trazado por hitos en la lógica y el análisis matemático. Desde Ada Lovelace, quien anticipó la capacidad de las máquinas para manipular símbolos y algoritmos, hasta el desarrollo del álgebra booleana por George Boole, fundamental para el diseño de circuitos y la programación lógica. La publicación de los Principia Mathematica por Bertrand Russell y Alfred Whitehead intentó demostrar que las matemáticas elementales podían reducirse a un razonamiento mecánico, un concepto que, aunque limitado por los teoremas de incompletitud de Kurt Gödel, sentó las bases del pensamiento computacional moderno, y hoy trasciende a la inteligencia artificial generativa.

A pesar del gran éxito de las aplicaciones de inteligencia artificial, existe una desconexión importante entre la facilidad de uso de las herramientas y la comprensión profunda de los principios matemáticos que las sostienen. Esta situación ha dado lugar a lo que varios expertos llaman la brecha de habilidades (skills gap), donde profesionales e ingenieros pueden implementar modelos complejos con librerías como Scikit-Learn, TensorFlow o PyTorch sin entender por qué funcionan o, aún más peligroso, bajo qué condiciones fallan.

Este problema tiene varias dimensiones que afectan la sostenibilidad y la ética en el desarrollo tecnológico. Primero, la dependencia de modelos de caja negra limita la innovación. Quienes no poseen una base matemática sólida se ven relegados a ser simples usuarios de algoritmos existentes, sin capacidad para crear variaciones originales o desarrollar nuevos paradigmas de aprendizaje que aborden problemas específicos. La verdadera ventaja competitiva en el mercado laboral de la inteligencia artificial no consiste en

saber instalar una librería, sino en tener un conocimiento profundo de los algoritmos y sus implementaciones para optimizarlos o mejorarlos.

Capítulo 1

Fundamentos matemáticos, modelización estocástica y estructuras avanzadas en el aprendizaje de máquina

El aprendizaje de máquina se sitúa en la intersección de las ciencias de la computación, la ingeniería de sistemas y la matemática aplicada. En las últimas décadas, la democratización de herramientas de software de código abierto y de librerías de alto nivel ha permitido a miles de profesionales aplicar modelos predictivos complejos sin requerir un conocimiento profundo de las ecuaciones subyacentes.

Sin embargo, la abstracción excesiva de estos detalles técnicos conlleva el riesgo de que los desarrolladores ignoren los límites de aplicabilidad y las decisiones de diseño inherentes a los algoritmos. Comprender los principios matemáticos no es simplemente un ejercicio académico, sino una necesidad imperativa para evitar fallos estructurales en la arquitectura de modelos, para diagnosticar comportamientos anómalos y para proponer innovaciones capaces de superar las limitaciones actuales de la inteligencia artificial

El estudio formal de las matemáticas para el aprendizaje de máquina abarca principalmente el álgebra lineal, el cálculo multivariable, la probabilidad y la estadística, y la teoría de la información Asimismo, fronteras más avanzadas de la investigación recurren a la geometría diferencial y la

topología para abordar estructuras de datos no euclidianas Este reporte proporciona una revisión exhaustiva de estos dominios, analizando cómo interactúan entre sí y cómo fundamentan los algoritmos modernos, además de evaluar el panorama pedagógico actual para la adquisición de estas competencias.

El continuo de la literatura especializada y la brecha de aplicación

Una problemática recurrente en los entornos de desarrollo es la desconexión entre la investigación teórica avanzada y la aplicación práctica en el ámbito de los negocios. Quienes se enfocan excesivamente en la publicación de artículos académicos suelen alejarse de los casos de uso comerciales, mientras que los profesionales que operan exclusivamente con herramientas de caja negra carecen de la capacidad de modificar o proponer variaciones algorítmicas sustanciales. Esta tensión dialéctica determina el tipo de literatura que cada perfil debe consultar para consolidar sus bases matemáticas (González, 2025).

En este sentido, los textos clásicos asumen que el lector posee un dominio consumado del cálculo y el álgebra lineal, relegando las matemáticas a apéndices breves. Por el contrario, trabajos más recientes se enfocan deliberadamente en la exposición didáctica de los prerrequisitos. La siguiente matriz comparativa sintetiza las ventajas relativas de los textos más citados en el ámbito académico y profesional, lo que permite al lector ponderar su elección según su formación de origen y sus aspiraciones técnicas (véase la Tabla 1).

Tabla 1: Ventajas relativas de los textos más citados en el ámbito académico y profesional

Título del libro	Autores Principales	Enfoque Predominante y Ventajas Distintivas
Mathematics for Machine Learning	Deisenroth, Faisal y Ong.	Diseñado explícitamente para cerrar la brecha formativa entre la educación secundaria y la universidad estilo matemático riguroso pero accesible, vinculando la teoría directamente con aplicaciones como PCA, regresión lineal, modelos de mezclas gaussianas y SVM
Pattern Recognition y Machine Learning	Christopher Bishop	Enfoque marcadamente probabilístico y bayesiano. Posee una exposición cuidadosa y minuciosa, con abundantes gráficos intuitivos para explicar derivaciones matemáticas complejas, lo que resulta ideal para principiantes con formación básica.
The Elements of Statistical Learning	Hastie, Tibshirani y Friedman	Se considera el estándar de oro en los fundamentos estadísticos clásicos del aprendizaje automático. Las formulaciones abstractas de alta rigurosidad y sus

		demostraciones matemáticas lo vuelven el texto idóneo para perfiles con madurez matemática previa
Deep Learning	Goodfellow, Bengio, y Courville.	Texto enfocado en los fundamentos del aprendizaje profundo. Posee una fuerte carga de cálculo multivariable, álgebra matricial y optimización, orientada específicamente a las redes neuronales artificiales.
Probabilistic Machine Learning: An Introduction	Kevin Patrick Murphy.	Funciona como una enciclopedia exhaustiva de métodos de aprendizaje de máquina. Su enfoque probabilístico abarca tanto métodos clásicos como representaciones no paramétricas avanzadas y variables latentes

La sugerencia didáctica predominante en las comunidades de aprendizaje indica comenzar con el texto de Deisenroth et al. para asentar las nociones de álgebra y cálculo antes de transicionar hacia los libros de Bishop o Goodfellow Marina Wyss, científica aplicada en Amazon, sugiere que no se debe percibir la matemática como una barrera inicial inaccesible; al contrario, propone construir proyectos prácticos primero, de modo que las necesidades matemáticas surjan de forma orgánica a medida que se resuelven problemas

reales de optimización o ajuste de hiper parámetros.

Fundamentos del álgebra lineal en espacios multidimensionales

El álgebra lineal constituye el andamiaje fundamental mediante el cual se estructuran y manipulan los datos en cualquier sistema de aprendizaje de máquina. Los objetos de estudio de esta disciplina, como escalares, vectores, matrices y tensores, sirven como contenedores primarios de información. En un conjunto de datos supervisado convencional, cada observación se modela como un vector en un espacio vectorial euclidiano de n dimensiones, mientras que la colección completa de datos de entrenamiento conforma una matriz X, vinculada a un vector de etiquetas o salidas esperadas y_5 (Moyano et al., 2024).

Las operaciones matriciales no solo simplifica la notación, sino que reflejan transformaciones geométricas tangibles del espacio de datos La multiplicación de matrices representa composiciones de transformaciones lineales que permiten proyectar datos de un espacio dimensional a otro, rotar representaciones o escalar magnitudes El producto punto entre dos vectores $\mathbf{u} \cdot \mathbf{v} = \sum u_i v_i$ mide la proyección de un vector sobre otro, revelando nociones de similitud que resultan críticas en los motores de búsqueda semántica y en el cálculo de mecanismos de atención en redes neuronales transformadoras.

Descomposiciones matriciales y reducción de

dimensionalidad

A medida que los conjuntos de datos aumentan en volumen y complejidad, el álgebra lineal proporciona herramientas para extraer las estructuras latentes dominantes de estos conjuntos. La descomposición en valores propios (eigendecomposition) y la descomposición en valores singulares (SVD) permiten desfactorizar matrices masivas en componentes ortogonales simplificados. En la ecuación característica de una matriz cuadrada, los valores propios λ cuantifican el factor de estiramiento o compresión que sufre el espacio a lo largo de las direcciones determinadas por los vectores propios correspondientes.

El Análisis de Componentes Principales (PCA) se basa directamente en la descomposición de la matriz de covarianza para proyectar los datos en un subespacio de menor dimensión, preservando la mayor cantidad posible de varianza. Esta técnica elimina la redundancia entre variables colineales, acelera el entrenamiento de modelos posteriores y mitiga la maldición de la dimensionalidad (Kitao, 2022). Los investigadores han destacado también la relevancia de profundizar en el cálculo matricial puro, dado que operaciones como la transposición y la inversión de matrices resultan ineludibles para derivar analíticamente los gradientes en redes densas.

Cálculo multivariable y optimización determinista

Si el álgebra lineal describe el estado y la geometría de los datos, el cálculo diferencial e integral describe el cambio y la optimización de los

modelos a lo largo del tiempo. El propósito cardinal de la mayoría de los algoritmos de aprendizaje de máquina es minimizar una función de pérdida o coste que cuantifica la discrepancia entre las predicciones del modelo y la realidad observada.

El cálculo diferencial introduce el concepto de derivada, que mide la tasa de cambio instantánea de una función. En espacios de alta dimensión, donde los modelos poseen millones de parámetros (como en el aprendizaje profundo), las derivadas parciales se agrupan en un vector llamado gradiente. El gradiente apunta en la dirección del máximo local de la función de coste. Por ende, para minimizar el error, los parámetros del modelo se actualizan iterativamente en la dirección opuesta al gradiente, un proceso conocido como descenso de gradiente.

La regla de la cadena del cálculo diferencial es el motor que impulsa el algoritmo de retropropagación (backpropagation) en las redes neuronales artificiales Este algoritmo permite calcular el gradiente de la función de pérdida con respecto a cada peso de la red, propagando el error desde la capa de salida hacia las capas iniciales Sin una comprensión rigurosa de la regla de la cadena y el comportamiento de las derivadas, resulta imposible diagnosticar problemas severos en el entrenamiento de redes profundas, tales como el desvanecimiento o la explosión del gradiente (Kim, 2025). Adicionalmente, el análisis de convexidad y la identificación de mínimos locales y globales ayudan a los investigadores a discernir por qué ciertos algoritmos convergen rápidamente, mientras que otros quedan atrapados en regiones subóptimas del paisaje de pérdida.

Estadística matemática y pruebas de hipótesis

El aprendizaje de máquina se enfrenta intrínsecamente a datos ruidosos, incompletos o inciertos. La estadística proporciona el marco para recolectar, describir e interpretar los datos, mientras que la teoría de la probabilidad permite modelar matemáticamente los procesos estocásticos que los generan.

El análisis predictivo requiere realizar suposiciones rigurosas sobre la distribución subyacente de las variables estudiadas El conocimiento profundo de las distribuciones discretas y continuas capacita a los profesionales para seleccionar la clase adecuada de modelos estadísticos Por ejemplo, asumir una distribución gaussiana (normal) en los residuos es una premisa básica de los modelos de regresión lineal por mínimos cuadrados, mientras que problemas que involucran el conteo de eventos discretos en intervalos de tiempo (como la llegada de usuarios a un servidor) requieren la aplicación de distribuciones de Poisson.

Más allá de la mera descripción de datos, la estadística inferencial es el mecanismo principal mediante el cual las empresas tecnológicas validan el rendimiento real de los modelos desplegados en producción Las pruebas A/B, en las que se compara un grupo de prueba frente a un grupo de control, dependen por completo del planteamiento de las pruebas de hipótesis y del cálculo de los intervalos de confianza La comprensión de los diferentes tipos de pruebas estadísticas permite a los ingenieros asegurarse de que las mejoras observadas en las métricas de precisión no se deben al azar.

La Tabla 2 presenta de manera concisa las pruebas de hipótesis más empleadas en la validación de modelos y la experimentación en ciencia de datos:

Tabla 2: Pruebas de hipótesis más empleadas en la validación de modelos

Prueba Estadística	Propósito Principal y Mecanismo	Aplicación común en aprendizaje de máquina
Prueba Z	Compara las medias poblacionales cuando la varianza poblacional es conocida y el tamaño de la muestra es grande.	Validación de métricas de precisión en flujos de datos masivos.
Prueba T (Student)	Compara las medias de dos grupos cuando la varianza es desconocida o la muestra es pequeña.	Pruebas A/B para evaluar el impacto de un nuevo algoritmo de recomendación.
Prueba chi-cuadrado	Evalúa la independencia entre variables categóricas o la bondad de ajuste.	Selección de características categóricas y análisis de correlación no lineal.
Prueba F	Compara las varianzas de dos poblaciones o evalúa múltiples factores simultáneamente	Empleada la prueba de varianza de un factor (ANOVA) para comparar múltiples arquitecturas de modelos.

La correcta aplicación de estos contrastes garantiza que la toma de decisiones basada en datos posea significancia estadística, mitigando el riesgo de sobreajuste o la adopción de modelos espurios

Teoría de la información y flujos de conocimiento en redes profundas

Introducida por Claude Shannon en 1948, la teoría de la información provee el andamiaje cuantitativo para medir la incertidumbre y el contenido informativo de variables aleatorias. Al seleccionar variables explicativas, optimizar arquitecturas neuronales o segmentar nodos en árboles de decisión, se aplican directamente los principios informacionales.

El concepto de piedra angular es la entropía de Shannon, que mide el grado de impredecibilidad o aleatoriedad de una distribución de probabilidad. Para distribuciones uniformes, en las que todos los resultados son igualmente probables, la entropía alcanza su máximo; para distribuciones sesgadas, en las que un resultado es altamente predecible, la entropía es baja. En algoritmos de inducción de árboles de decisión como ID3 o CART, la ganancia de información (basada en la reducción de la entropía) determina qué variable predictora ofrece la mejor partición del conjunto de datos en cada nodo.

Otra métrica insustituible es la divergencia de Kullback-Leibler (KL), que cuantifica la cantidad de información adicional necesaria para codificar eventos de una distribución verdadera mediante una distribución aproximada propuesta por un modelo. Minimizar la divergencia KL entre la distribución predicha por el modelo y la distribución empírica de los datos es la base de la optimización de múltiples algoritmos generativos modernos, tales como los decodificadores variacionales (VAE), que regularizan el espacio latente, asegurando que este siga una distribución normal estándar.

La relación matemática intrínseca entre estos conceptos se manifiesta en la pérdida de entropía cruzada (cross-entropy loss), ampliamente utilizada para entrenar clasificadores neuronales multiclase. La entropía cruzada combina la incertidumbre propia de los datos (la entropía original) y la

penalización por la ineficiencia de la aproximación algorítmica (la divergencia KL), lo que constituye un objetivo directo para el descenso de gradiente (Yolles, 2022).

Investigaciones contemporáneas sugieren que las redes neuronales profundas operan según el principio de embotellamiento de información (information bottleneck). Según esta teoría, en las primeras capas de la red se codifica la totalidad de la información de entrada, mientras que las capas subsecuentes comprimen progresivamente dicha información, eliminando las características irrelevantes y preservando únicamente las representaciones abstractas necesarias para realizar la predicción final.

No obstante, investigadores del Instituto Santa Fe descubrieron que para tareas de clasificación comunes y deterministas, las medidas convencionales de embotellamiento de información pueden comportarse de manera no intuitiva, agrupando conceptos disímiles de forma trivial. Esto subraya que la comunidad científica aún busca métricas de comprensión que reflejen fielmente la abstracción humana de alto nivel.

Geometría diferencial y topología algebraica en datos no euclidianos

El legado duradero de la geometría euclidiana clásica subyace al aprendizaje de máquina tradicional, que asume que los datos residen en espacios euclidianos. No obstante, los desarrollos de frontera en inteligencia artificial se enfrentan cada vez más a datos inherentemente no euclidianos y ricamente estructurados, que presentan geometrías curvas, geometrías complejas o interacciones relacionales en grafos. Para extraer conocimiento

útil de tales estructuras, se requiere una perspectiva matemática mucho más amplia que recurra a la geometría diferencial, el álgebra abstracta y la topología algebraica

El enfoque de las variedades diferenciables en espacios curvos

La geometría diferencial estudia las propiedades de las curvas y superficies mediante el cálculo y el álgebra lineal. En el contexto del aprendizaje de máquina, tanto el espacio de datos como el de los parámetros de una red neuronal pueden tratarse matemáticamente como variedades (manifolds). Una variedad es un espacio topológico que, localmente, se asemeja al espacio euclidiano plano, pero que, a escala global, puede presentar una curvatura intrincada (Moyano et al., 2024).

Por ejemplo, la llamada geometría de la información dota al espacio de las distribuciones de probabilidad de una estructura métrica riemanniana mediante la métrica de Fisher-Rao Al entrenar un modelo, avanzar en línea recta en el espacio de parámetros euclidiano no siempre implica un cambio uniforme en el comportamiento del modelo; en su lugar, el descenso de gradiente natural propone avanzar siguiendo las geodésicas (las rutas más cortas sobre la superficie curva) de la variedad de distribuciones de probabilidad, acelerando sustancialmente la convergencia algorítmica y dotando de mayor robustez teórica a la optimización Investigaciones doctorales recientes demuestran además el valor de la geometría de Finsler en la comparación de longitudes esperadas derivadas de variedades estocásticas, ofreciendo generalizaciones que sobrepasan el marco clásico de Riemann.

Análisis de datos topológicos en imagenología médica

Mientras que la geometría diferencial realiza mediciones precisas de distancias y ángulos en superficies, la topología adopta un enfoque mucho más abstracto y flexible. La topología estudia las propiedades de los espacios que permanecen invariantes bajo deformaciones continuas (estiramientos, torsiones), prescindiendo de nociones rígidas de distancia métrica y centrándose únicamente en la conectividad y la vecindad entre puntos.

El Análisis de Datos Topológicos (TDA) utiliza herramientas de la topología algebraica, como la homología persistente, para capturar descriptores globales de conjuntos de datos de alta dimensión. El TDA es capaz de detectar la presencia de agujeros, cavidades y componentes conectados en las nubes de puntos de datos, estructuras que los algoritmos estadísticos convencionales suelen ignorar (Gallego et al., 2025).

Se ha comprobado que la extracción previa de características topológicas y geométricas, antes de suministrar los datos a un modelo de aprendizaje profundo, puede reducir drásticamente el tiempo de entrenamiento y elevar sustancialmente el rendimiento de la red en radiómica, patómica y técnicas multiómicas, lo que contribuye de forma decisiva al arsenal de la medicina de precisión.

Estrategias pedagógicas, cursos masivos y curvas de aprendizaje

Ante el volumen y la abstracción de las disciplinas matemáticas requeridas, la brecha entre el nivel cubierto en la educación secundaria y la

madurez matemática exigida por los libros de texto avanzados de aprendizaje de máquina representa un desafío formidable para estudiantes y profesionales autodidactas Marc Peter Deisenroth propone una analogía musical sumamente esclarecedora para categorizar la manera en que las personas interactúan con el aprendizaje de máquina y determinar el nivel de profundidad matemática necesario para cada una:

1. **El Oyente Astuto**: Representa a los usuarios finales o expertos de dominio no técnicos que se benefician de herramientas empaquetadas de código abierto y de servicios en la nube, sin preocuparse por los detalles algorítmicos profundos. Este perfil requiere únicamente una intuición conceptual básica.
2. **El Intérprete o Practicante Estereotípico**: Es el científico de datos o ingeniero de software que comprende las interfaces y los casos de uso y es capaz de realizar proezas predictivas mediante la manipulación de modelos existentes. Para ellos, la comprensión matemática les permite discernir los límites y los beneficios de cada método y aplicarlos con cierta flexibilidad.
3. **El Compositor Novel**: Son los investigadores que deben proponer y explorar enfoques novedosos para el aprendizaje de datos, requiriendo un dominio analítico absoluto de las bases subyacentes para formular nuevas estructuras y funciones de pérdida

En respuesta a estas divergencias de perfiles, el ecosistema educativo ha ofrecido una amplia variedad de cursos masivos en línea. El equilibrio entre la teoría pura y la traducción de ecuaciones en código ejecutable constituye la principal distinción entre los programas ofrecidos por universidades de élite y

las especializaciones industriales. La Tabla 3 expone detalladamente los cursos más reconocidos para la adquisición de estas competencias:

Tabla 3: Cursos más reconocidos para la adquisición de competencias digitales

Institución / Plataforma	Nombre del Curso	Duración	Enfoque y Características Distintivas
DeepLearning.AI	Math for ML & Data Science.	12 - 13 semanas.	Nivel principiante orientado a desarrolladores autodidactas Emplea un enfoque visual para romper barreras matemáticas, implementando algoritmos de clasificación y optimización directamente en código ejecutable de Python
Imperial College London	Mathematics for Machine Learning.	17 semanas.	Nivel intermedio que asume familiaridad previa con demostraciones matemáticas formales y programación en NumPy Actúa como un puente

			hacia la investigación en IA, implementando algoritmos desde cero
MIT OpenCourseWare	18.06 Linear Algebra (Prof. Gilbert Strang).	36 horas.	Clase maestra en pensamiento matemático y computacional se enfoca en preparar a los alumnos avanzados para la lectura de artículos científicos de frontera, cubriendo descomposiciones matriciales profundas
Udemy	Math Foundations of ML.	16 horas.	Orientado a la implementación práctica, demostrando línea por línea cómo las fórmulas abstractas se convierten en código en marcos de trabajo como NumPy, TensorFlow y PyTorch

MIT OpenLearning	Math of Big Data and ML	14 horas.	Nivel avanzado enfocado en resolver restricciones de escala masiva de petabytes, utilizando álgebra de matrices asociativas e integrando bases de datos con teoría de grafos
Columbia University / edX	Essential Math for AI.	6 horas.	Funciona como un repaso relámpago e intensivo para profesionales que necesitan evaluar sus lagunas de conocimiento antes de entrevistas técnicas rigurosas

Herramientas interactivas y entornos visuales de experimentación

Paralelamente a los cursos estructurados, el paradigma educativo en inteligencia artificial ha girado con fuerza hacia los entornos de exploración interactivos ejecutados directamente en el navegador web.· Estas plataformas rompen la aridez de las ecuaciones estáticas y permiten a los estudiantes e investigadores forjar una comprensión táctil y dinámica de los flujos de

información en los modelos de aprendizaje de máquina.

La Tabla 4 compendia las herramientas visuales e interactivas de código abierto más destacadas desarrolladas recientemente por la comunidad científica y académica:

Tabla 4: Herramientas visuales e interactivas de código abierto

Nombre de la herramienta	**Categoría y Enfoque**	**Descripción y Capacidades Operativas**
Transformer Explainer	Modelos de Lenguaje	Ejecuta un modelo GPT-2 en tiempo real en el navegador para que los usuarios experimenten con texto y observen en tiempo real la predicción de tokens y el funcionamiento de las capas de atención.
CNN Explainer	Visión Artificial	Diseñado para ayudar a no expertos a comprender las redes neuronales convolucionales mediante la visualización e inspección directa de sus capas operativas.

BertViz	Interpretabilidad	Extiende las herramientas de visualización para proyectar la atención interna que prestan los modelos masivos basados en transformadores como BERT, GPT-2 y RoBERTa.
exBERT	Análisis de NLP	Ayuda a los humanos a realizar investigaciones interactivas y a formular hipótesis sobre el proceso de deducción interna en modelos de procesamiento del lenguaje natural.
GAN Lab	Modelos Generativos	Permite explorar redes generativas antagónicas (GANs), controlar manualmente los hiperparámetros y observar la ejecución en cámara lenta en el navegador.

Embedding Projector	Álgebra Lineal	Ayuda a entender cómo los modelos interpretan los datos mediante representaciones vectoriales (embeddings), lo que permite explorar algoritmos como PCA, t-SNE y UMAP.
The Language Interpretability Tool (LIT)	Ética y Sesgos	Plataforma abierta de visualización para comprender modelos de NLP, orientada a descubrir comportamientos adversariales, falta de consistencia y sesgos de género o estilo
What-If Tool	Auditoría de Modelos	Permite probar visualmente el comportamiento de los modelos de aprendizaje de máquina ya entrenados con un mínimo de código.

Seeing Theory	Probabilidad y Estadística	Creado originalmente en la Universidad de Brown Enseña conceptos estocásticos y bayesianos mediante manipulables táctiles interactivos en el navegador

La revisión exhaustiva realizada demuestra que las matemáticas aplicadas al aprendizaje de máquina no constituyen un conjunto de fórmulas aisladas, sino un continuo analítico integrado en el que cada disciplina desempeña un rol sinérgico insustituible. El álgebra lineal provee los cimientos para organizar flujos masivos de datos en espacios de múltiples dimensiones.

El cálculo multivariable dinamiza dichos datos, posibilitando la optimización paramétrica mediante formulaciones precisas del gradiente y de la regla de la cadena. Por su parte, la estadística matemática dota al sistema de la capacidad de operar bajo incertidumbre, de modelar ruidos y de emitir juicios predictivos respaldados por pruebas de hipótesis rigurosas, indispensables en el entorno empresarial (Ríos y de la Fuente, 2023). Finalmente, la teoría de la información establece los límites de compresión y los flujos de entropía indispensables para el entrenamiento de redes masivas.

El advenimiento de arquitecturas que superan las limitaciones de los espacios continuos planos exige una creciente alfabetización en geometría diferencial y topología algebraica, campos que ya demuestran su valor en la medicina de precisión y en el procesamiento de señales no euclidianas La

desconexión entre el uso utilitario de librerías de software de caja negra y la ignorancia de los fundamentos matemáticos subyacentes representa una vulnerabilidad operativa significativa para la industria tecnológica actual Por consiguiente, la inmersión estructurada en estos lenguajes analíticos, impulsada y facilitada por la profusión de entornos de visualización interactiva y cursos de traducción a código, se perfila como la única vía sostenible para aquellos profesionales que aspiren a convertirse en compositores y líderes de las arquitecturas del mañana en el aprendizaje de máquina

Capítulo 2
Álgebra lineal para machine learning

El álgebra lineal se erige como la columna vertebral matemática de múltiples disciplinas científicas, pero su relevancia en el campo del aprendizaje automático y la inteligencia artificial no tiene parangón. No se trata simplemente de un conjunto de herramientas operativas para agilizar cálculos numéricos, sino del lenguaje fundamental mediante el cual se estructuran, manipulan y transforman los datos en representaciones abstractas que los algoritmos computacionales pueden procesar y comprender de manera óptima.

En la era del procesamiento masivo de información y el aprendizaje profundo, los conjuntos de datos rara vez se presentan como variables aisladas o simples listas unidimensionales; por el contrario, la información del mundo real se caracteriza por su alta dimensionalidad y su complejidad estructural Desde la representación geométrica de imágenes compuestas por millones de píxeles hasta la modelización de la semántica en el procesamiento del lenguaje natural mediante incrustaciones de palabras, el álgebra lineal proporciona el andamiaje necesario para operar en estos vastos espacios multidimensionales.

La conexión histórica y conceptual de esta disciplina con el desarrollo de algoritmos computacionales data de siglos atrás, cuando matemáticos como Leonhard Euler en el siglo XVIII idearon métodos para resolver sistemas de ecuaciones lineales aplicados a la astronomía y la física Posteriormente, en

la segunda mitad del siglo XIX, figuras como Arthur Cayley y James Joseph Sylvester consolidaron el álgebra matricial, sentando las bases teóricas que hoy en día permiten a los científicos de datos resolver problemas de optimización a gran escala En el panorama contemporáneo, el álgebra lineal permite expresar conjuntos de datos complejos de una forma que los algoritmos pueden entender y procesar, lo que posibilita construir modelos de alta complejidad utilizando una gran cantidad de datos recopilados del mundo real.

Estructuras fundamentales de datos y su representación geométrica

Para comprender la profundidad con la que el álgebra lineal impregna el aprendizaje automático, resulta imperativo examinar las estructuras de datos fundamentales que componen cualquier arquitectura analítica. En el nivel más básico, se encuentran los escalares, que representan magnitudes individuales sin dirección, como un único número real que podría denotar la tasa de aprendizaje de un algoritmo o un parámetro de regularización (Guzmán et al., 2025). Sin embargo, la unidad de procesamiento por excelencia en el machine learning es el vector, concebido como una matriz ordenada de números que puede interpretarse tanto de manera algebraica como geométrica.

Geométricamente, un vector representa un punto o una flecha en un espacio n-dimensional; algebraicamente, encapsula las características o atributos de una sola observación dentro de un conjunto de datos Por ejemplo, en un sistema de tasación inmobiliaria, un vector puede contener las

dimensiones de una vivienda, el número de habitaciones y la distancia al centro urbano Cuando múltiples vectores de características se agrupan para representar un conjunto completo de datos, emerge la matriz Una matriz bidimensional dispone las observaciones en sus filas y las variables o características en sus columnas, estableciendo una cuadrícula numérica que permite realizar operaciones masivas de manera simultánea. Esta estructura no solo es eficiente para el almacenamiento, sino que también sirve como operador de transformaciones lineales, un concepto medular en el que una matriz actúa sobre un vector para rotarlo, escalarlo o proyectarlo en un nuevo espacio vectorial.

Finalmente, los tensores generalizan los conceptos de escalares, vectores y matrices a dimensiones superiores. En el aprendizaje profundo, los tensores son el estándar absoluto para alimentar información a las redes neuronales artificiales Una imagen en color, por ejemplo, no puede representarse de manera óptima en una matriz bidimensional simple si se desea preservar la información de los canales cromáticos; por lo tanto, se utiliza un tensor tridimensional donde la altura, la anchura y los canales de color forman tres ejes coordinados separados. La capacidad de generalizar estas estructuras permite a los modelos procesar flujos continuos de datos masivos sin perder la coherencia espacial y estructural de la información de origen.

La Tabla 5 sintetiza estas estructuras algebraicas y su correspondencia directa dentro de los flujos de trabajo tradicionales en la ciencia de datos y el aprendizaje automático:

Tabla 5: Correspondencia directa entre estructuras algebraicas y flujos de trabajo convencionales en ciencia de datos y aprendizaje automático.

Estructura Algebraica	**Definición Matemática**	**Interpretación en Machine Learning**	**Dimensión Típica**
Escalar	Un único valor numérico real sin dirección asociada.	Parámetros del modelo, tasas de aprendizaje, coeficientes de sesgo.	0-D
Vector	Arreglo ordenado de números dispuesto en una sola columna o fila.	Vector de características de una observación o de coordenadas espaciales.	1-D
Matriz	Arreglo bidimensional de números, organizado en filas y columnas.	Conjunto de datos tabulares completo (filas como muestras y columnas como variables).	2-D

Tensor	Generalización multidimensional de matrices para n ejes.	Imágenes multicanales, secuencias de video y procesamiento por lotes en redes neuronales.	n-D ($n \geq 3$)

La comprensión de estas estructuras permite a los investigadores abstraer la naturaleza física de los datos. No importa si la entrada original corresponde a señales de audio, archivos de texto o telemetría satelital; una vez codificados en forma de vectores y matrices, todos los datos se someten a los mismos principios y reglas de procesamiento matemático. Esta unificación algorítmica ha permitido el vertiginoso avance de la inteligencia artificial en campos aparentemente inconexos.

Operaciones algebraicas y su rol en la mecánica del aprendizaje

Dentro del nutrido repertorio de operaciones que el álgebra lineal aporta al aprendizaje automático, el producto punto y la multiplicación de matrices destacan como catalizadores fundamentales de la inferencia y el entrenamiento. El producto punto entre dos vectores no es simplemente una operación aritmética que produce un escalar, sino que tiene una interpretación geométrica de gran calado: mide la proyección de un vector sobre otro y, por ende, su similitud direccional (De la Cruz, 2025).

En los sistemas de recomendación modernos, como los implementados

por plataformas de transmisión de video o de comercio electrónico, el producto punto se utiliza ampliamente para calcular la alineación entre el vector que representa las preferencias latentes de un usuario y el que describe los atributos de un producto determinado. Una puntuación elevada en el producto punto indica una alta afinidad geométrica, lo que se traduce en una recomendación precisa de contenido.

La multiplicación de matrices, por su parte, representa la composición de transformaciones lineales. Cuando un vector de entrada atraviesa una capa de una red neuronal artificial, se somete esencialmente a una multiplicación matricial con una matriz de pesos aprendidos. Esta operación proyecta los datos de entrada en un nuevo espacio de características, donde las relaciones complejas pueden volverse linealmente separables o más fáciles de procesar en capas posteriores.

Resulta fascinante observar que si no fuera por la adición de funciones de activación no lineales entre estas multiplicaciones matriciales, cualquier red neuronal profunda colapsaría matemáticamente en un modelo de regresión lineal simple, dado que la composición de múltiples transformaciones lineales consecutivas equivale invariablemente a una única transformación lineal. A continuación se presenta la tabla 6 de operaciones fundamentales en el álgebra lineal computacional y su impacto en la construcción de modelos:

Tabla 6: Operaciones esenciales del álgebra lineal computacional y su influencia en el desarrollo de modelos.

Operación Algebraica	**Definición Formal**	**Aplicación Práctica en**

		Algoritmos
Producto Punto	$\mathbf{u} \cdot \mathbf{v} = \sum u_i v_i$	Medición de la similitud en sistemas de recomendación y cálculo de las activaciones.
Multiplicación Matricial	$\mathbf{C} = \mathbf{AB}$	Propagación hacia adelante en redes neuronales e ingeniería de características masiva.
Matriz Transpuesta	$\mathbf{A}^T$ (intercambio de filas por columnas)	Alineación de dimensiones para operaciones de multiplicación y de derivación de gradientes.
Matriz Inversa	$\mathbf{A}^{-1}$ tal que $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$	Solución analítica de las ecuaciones normales en la regresión lineal clásica.

El uso de bibliotecas optimizadas como NumPy permite realizar estas operaciones sin recurrir a bucles iterativos tradicionales en lenguajes de alto nivel, lo que resultaría prohibitivamente lento con grandes volúmenes de datos. En su lugar, estas bibliotecas delegan el cálculo a rutinas de bajo nivel altamente optimizadas que aprovechan las capacidades de vectorización del

hardware y la paralelización en procesadores modernos y unidades de procesamiento gráfico (Raschka et al., 2020).

Un mecanismo crucial en el álgebra lineal aplicada a la computación es el concepto de transmisión o broadcasting. Este mecanismo permite a las bibliotecas computacionales operar con arreglos de distintas formas en operaciones aritméticas, lo que suele hacer que el código sea más conciso y rápido. Por ejemplo, si se desea sumar un vector constante a cada fila de una matriz de mayor dimensión, las reglas de transmisión permiten expandir virtualmente el operando de menor rango sin necesidad de replicar físicamente los datos en la memoria, lo que optimiza drásticamente los recursos computacionales durante la fase de entrenamiento.

Las reglas estrictas de la transmisión requieren que dos dimensiones sean compatibles si tienen la misma longitud o si una de ellas es igual a la otra[1]. En cualquier dimensión en la que un arreglo tenga tamaño 1 y el otro tenga un tamaño mayor que 1, el primer arreglo se comporta como si fuera copiado a lo largo de esa dimensión. Este fenómeno es el núcleo que permite a los algoritmos procesar conjuntos masivos de datos, manteniendo una huella de memoria baja.

Sistemas de ecuaciones lineales y optimización de modelos

Muchos de los algoritmos fundamentales en el aprendizaje automático pueden formularse en última instancia como la búsqueda de soluciones para sistemas de ecuaciones lineales. El ejemplo arquetípico es la regresión lineal,

un método estadístico utilizado para predecir el valor de una variable dependiente continua a partir de una o más variables independientes o predictoras. En un escenario de regresión lineal múltiple con n características, el objetivo es encontrar un vector de pesos $\mathbf{w}$ y un sesgo b que satisfagan de la mejor manera posible la ecuación $y = \mathbf{w}^T\mathbf{x} + b$ para todas las observaciones del conjunto de datos.

Cuando se dispone de un conjunto masivo de datos de entrenamiento, el sistema de ecuaciones resultante suele estar sobre determinado, lo que significa que existen más ecuaciones que incógnitas y, por tanto, no existe una solución exacta que satisfaga todas las igualdades debido al ruido intrínseco de los datos del mundo real En este punto, el álgebra lineal introduce el concepto de mínimos cuadrados En lugar de buscar una solución exacta que no existe, los algoritmos buscan el vector de coeficientes que minimice la norma de la diferencia entre las predicciones del modelo y los valores reales observados La solución analítica a este problema se conoce como la ecuación normal y se expresa matemáticamente como $\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$.

Para determinar analíticamente la viabilidad de resolver estos sistemas, se recurre a conceptos como el rango de una matriz y el determinante. El determinante de una matriz cuadrada indica si la matriz es invertible; si el determinante es cero, la matriz es singular y no posee inversa, lo que imposibilita el uso directo de la ecuación normal (Stanimirović et al., 2024).

Por otro lado, el rango describe el número máximo de columnas linealmente independientes de una matriz, lo que, a su vez, determina la dimensión del espacio vectorial generado por sus columnas. En machine

learning, si el rango es menor que el número de características, indica que hay redundancia en los datos recolectados, lo que requiere técnicas de eliminación o de regularización para evitar fallas catastróficas durante el entrenamiento.

La resolución algorítmica de estos sistemas implica técnicas de eliminación gaussiana, que transforman una matriz en su forma escalonada por filas mediante operaciones elementales de fila. Este proceso comprende la eliminación hacia adelante para generar ceros por debajo de la diagonal y la sustitución posterior hacia atrás para obtener las variables finales. Dado que el cálculo numérico por computadora es susceptible de errores de redondeo, se aplican estrategias como el pivoteo para evitar la división por cero y mejorar la estabilidad numérica global del algoritmo.

Sin embargo, la computación directa de la matriz inversa conlleva desafíos numéricos de gran magnitud. Si el conjunto de datos presenta multicolinealidad, es decir, si dos o más variables predictoras están altamente correlacionadas entre sí, la matriz de covarianza se vuelve casi singular o mal condicionada. Intentar invertir una matriz en tales condiciones produce errores numéricos masivos que desestabilizan por completo el entrenamiento del modelo y destruyen su capacidad de generalización.

Para mitigar este problema, los profesionales del aprendizaje automático recurren a la pseudoinversa de Moore-Penrose, calculada habitualmente mediante la descomposición en valores singulares, lo que permite encontrar una solución de mínimos cuadrados incluso cuando la matriz de covarianza no tiene rango completo, ignorando de manera efectiva las dimensiones redundantes que introducen el ruido numérico.

Descomposiciones matriciales como núcleo de

la reducción de dimensionalidad

A medida que los conjuntos de datos crecen en escala, a menudo sufren la maldición de la dimensionalidad. Cuando el número de variables o características de un conjunto de datos es muy elevado (a veces hasta alcanzar los miles o los millones), los puntos tienden a dispersarse exponencialmente en el espacio multidimensional. Esta dispersión hace que las métricas tradicionales de distancia pierdan efectividad, degrada el rendimiento computacional de los algoritmos y expone a los modelos a un riesgo severo de sobreajuste o overfitting, en el que la arquitectura de aprendizaje simplemente memoriza el ruido de entrenamiento en lugar de abstraer patrones subyacentes significativos (Hou y Behdinan, 2022).

La reducción de dimensionalidad es el proceso de proyectar datos desde un espacio de alta dimensionalidad a un subespacio de dimensión inferior, con el objetivo de preservar la mayor cantidad posible de la estructura geométrica original y de la varianza de la información. El álgebra lineal proporciona la maquinaria matemática necesaria para realizar estas reducciones mediante diversos métodos de descomposición matricial. La descomposición matricial consiste esencialmente en dividir una matriz compleja en el producto de matrices más simples y estructuradas que revelan las propiedades latentes del sistema de datos original

Existen varios métodos de descomposición recurrentes en el aprendizaje automático, cada uno con propiedades específicas que los hacen aptos para diferentes contextos analíticos y computacionales (véase la Tabla 7):

Tabla 7: Métodos de descomposición recurrentes en el aprendizaje automático

Método de Descomposición	**Descripción y Estructura Matemática**	**Aplicación en Machine Learning**
Descomposición LU	Divide una matriz en el producto de una matriz triangular inferior ($\mathbf{L}$) y otra superior ($\mathbf{U}$).	Resolución ágil y eficiente de sistemas de ecuaciones lineales y cálculo de determinantes
Descomposición QR	Factoriza una matriz en una matriz ortogonal ($\mathbf{Q}$) y una matriz triangular superior ($\mathbf{R}$).	Utilizado de forma predilecta para resolver problemas de mínimos cuadrados y computar valores propios
Descomposición en Valores Propios	Descompone una matriz diagonalizable en términos de sus vectores propios y valores propios ($\mathbf{A} =$).	Base matemática del Análisis de Componentes Principales para capturar la varianza máxima
Descomposición en Valores Singulares (SVD)	Factoriza cualquier matriz rectangular como el producto $\mathbf{U\Sigma V}^T$, donde $\mathbf{U}$ y $\mathbf{V}$ son ortogonales.	Reducción de dimensionalidad robusta, compresión de imágenes y modelado de temas latentes en NLP

El Análisis de Componentes Principales (PCA) es la técnica de reducción de dimensionalidad más extendida y se fundamenta directamente en la

descomposición espectral, o de valores propios, de la matriz de covarianza de los datos. Los vectores propios de la matriz de covarianza indican las direcciones espaciales en las que los datos presentan la mayor dispersión o varianza, mientras que los valores propios asociados cuantifican la magnitud de dicha varianza en cada una de esas direcciones (Vinegoni et al., 2020). Al ordenar los vectores propios en función de la magnitud decreciente de sus valores propios y conservar únicamente los primeros k componentes, es posible reducir sustancialmente las dimensiones del conjunto de datos, manteniendo intactas las relaciones estructurales de mayor peso informativo.

Por otro lado, la descomposición en valores singulares (SVD) se presenta como una alternativa aún más versátil y numéricamente estable que la descomposición de valores propios tradicional. A diferencia de esta última, que exige que la matriz sea cuadrada y diagonalizable, la SVD es aplicable a cualquier matriz rectangular$m \times n$, lo que la hace universalmente aplicable a matrices de datos crudos. Matemáticamente, la SVD descompone una matriz $\mathbf{A}$ en el producto $\mathbf{U\Sigma V}^T$, donde $\mathbf{\Sigma}$ es una matriz diagonal que contiene los valores singulares dispuestos en orden descendente.

La relación entre SVD y PCA es íntima. En términos computacionales, la SVD se utiliza a menudo como el motor numérico subyacente para calcular los componentes principales del PCA, evitando la formación explícita de la matriz de covarianza, lo cual puede resultar costoso e inestable en términos de precisión de coma flotante. Una de las grandes virtudes de la SVD es que permite cuantificar con precisión la pérdida de información que sufre un modelo al reducir sus dimensiones.

Dado que la varianza total de los datos equivale a la traza de la matriz de covarianza muestral, y esta, a su vez, es proporcional a la suma de los cuadrados de los valores singulares, la eliminación de los valores singulares más pequeños permite conocer con precisión matemática el porcentaje de varianza que se ha sacrificado en aras de la eficiencia computacional y la simplificación estructural.

Aparte de PCA y SVD, existen otros métodos de descomposición matricial no lineales o sujetos a restricciones específicas. La factorización matricial no negativa (NMF) asume que tanto los datos de entrada como las matrices factorizadas resultantes deben contener únicamente elementos no negativos. Esta aproximación es profundamente útil en el procesamiento de imágenes o en el análisis de texto, donde las intensidades de píxel o las frecuencias de palabras no pueden ser negativas por definición física. La NMF minimiza la distancia entre la matriz original y el producto de dos matrices no negativas, utilizando típicamente la norma de Frobenius al cuadrado como función de distancia de referencia.

En paralelo, el análisis de componentes independientes (ICA) se utiliza tradicionalmente para separar señales mixtas, un problema conocido en el procesamiento de señales como la separación ciega de fuentes. A diferencia de PCA, que busca direcciones ortogonales que maximicen la varianza (garantizando que las propiedades resultantes no estén correlacionadas entre sí), el ICA busca componentes estadísticamente independientes, modelando distribuciones que a menudo no son gaussianas.

El álgebra lineal en el paradigma del

aprendizaje profundo

El auge contemporáneo de las tecnologías de inteligencia artificial se debe en gran medida al éxito de las redes neuronales profundas y las arquitecturas de transformadores, si bien, conceptualmente, estos modelos imitan la transmisión de señales bioeléctricas entre neuronas biológicas, desde una perspectiva estrictamente computacional no son más que un ensamblaje masivo y jerárquico de operaciones de álgebra lineal acopladas a funciones de mapeo no lineales.

En el aprendizaje profundo, toda la información se representa y procesa en forma de tensores. En una red neuronal convolucional estándar orientada al reconocimiento de imágenes, los filtros de convolución se deslizan sobre el tensor tridimensional de entrada, realizando de forma iterativa productos punto de matrices pequeñas para extraer jerarquías de características espaciales, como bordes, texturas y formas complejas (Gao et al., 2025). A medida que el flujo de información avanza hacia las capas más profundas, las dimensiones espaciales suelen reducirse mediante operaciones de submuestreo, mientras que la profundidad de los canales de características aumenta, lo que permite a la red abstraer conceptos cada vez más semánticos y menos ligados a la cuadrícula de píxeles original.

El entrenamiento de estas inmensas estructuras se basa en el algoritmo de retropropagación o backpropagation. Después de que un lote de datos de entrenamiento fluye hacia adelante a través de la red y genera una predicción, una función de pérdida evalúa matemáticamente la desviación entre dicha predicción y la etiqueta real u objetivo Para minimizar este error, el algoritmo debe calcular cómo contribuyó cada uno de los millones o miles de millones

de parámetros (pesos y sesgos) de la red al error total.

Este cálculo masivo se resuelve aplicando la regla de la cadena del cálculo diferencial multivariable, pero su expresión y su ejecución se realizan enteramente mediante el cálculo matricial y las leyes de la álgebra lineal. El gradiente de la función de pérdida respecto a los pesos de una capa se calcula mediante el producto de matrices que contienen las derivadas parciales de la capa anterior y los estados de activación actuales. De este modo, la optimización de una red profunda se reduce a una secuencia orquestada de gigantescas multiplicaciones de matrices y operaciones de transposición que se repiten cíclicamente durante millones de iteraciones hasta alcanzar la convergencia del modelo.

La Tabla 8 presenta un desglose de las cuatro fases lógicas de la computación en una red neuronal artificial y su traducción matemática directa:

Tabla 8: Fases lógicas de una red neuronal artificial y su traducción matemática.

Fase de Entrenamiento	**Operación Matemática Subyacente**	**Rol de las Matrices y Vectores**
Propagación hacia adelante	$\mathbf{z} = \mathbf{W}\mathbf{x} + \mathbf{b}$	El vector de entrada se multiplica por la matriz de pesos para proyectarlo al nuevo espacio.

Función de Activación	$\mathbf{a} = \sigma(\mathbf{z})$	Operación no lineal, elemento a elemento, aplicada al vector resultante de la transformación.
Cálculo de Error	$\mathcal{L}(\mathbf{a}, \mathbf{y})$	Evaluación de la divergencia entre la predicción y el suelo real mediante normas vectoriales.
Retropropagación	$\nabla \mathcal{L} = \frac{\partial \mathcal{L}}{\partial \mathbf{W}}$	El uso de la regla de la cadena para propagar los errores hacia atrás mediante productos matriciales.

El vertiginoso progreso del aprendizaje profundo en la última década ha estado directamente impulsado por el desarrollo paralelo de hardware especializado capaz de ejecutar estas operaciones tensoriales masivas a velocidades vertiginosas, lo que permite entrenar en días modelos que antes habrían requerido siglos de computación secuencial tradicional.

Aplicaciones específicas en el procesamiento de texto e imágenes

El álgebra lineal no solo sienta las bases de los modelos, sino que también define cómo las máquinas interpretan los objetos físicos del mundo

real. En el ámbito de la visión computacional, una imagen digital no es más que una matriz bidimensional (para imágenes en escala de grises) o un tensor tridimensional (para imágenes a color) de valores numéricos que representan la intensidad de la luz en cada píxel (Oviedo y Jiménez, 2022). En consecuencia, operaciones geométricas comunes como la rotación, el escalado y la traslación de imágenes se implementan directamente mediante la aplicación de matrices de transformación lineal a las coordenadas de los píxeles.

Por ejemplo, en el entrenamiento de redes neuronales convolucionales aplicadas a imágenes, la aplicación de matrices de rotación al conjunto de datos de entrenamiento permite generar nuevas muestras sintéticas, lo que obliga al modelo a aprender a reconocer objetos independientemente de su orientación espacial, lo cual incrementa notablemente la robustez del sistema final ante variaciones imprevistas.

Por otro lado, en el procesamiento del lenguaje natural (NLP), los mayores retos provienen de la naturaleza categórica e inestructurada del lenguaje humano. Para que un algoritmo de machine learning procese texto, las palabras o frases individuales de un vocabulario deben mapearse a vectores de números reales mediante un proceso conocido como incrustación de palabras o word embedding.

Este procedimiento traslada la semántica de las palabras a relaciones de proximidad espacial en un hiperespacio de alta dimensión. Por ejemplo, palabras con significados o usos contextuales similares se posicionarán en vectores cuyas direcciones en el espacio multidimensional estén muy próximas, lo que permite que una simple operación de producto punto mida con precisión la correlación semántica entre dos oraciones complejas.

Adicionalmente, técnicas clásicas de modelado de temas como la Asignación Latente de Dirichlet (LDA) o el Análisis Semántico Latente (LSA) se apoyan en el álgebra lineal para extraer temáticas subyacentes de grandes corpus de texto Mediante la descomposición de matrices dispersas que cuantifican la frecuencia de términos en miles de documentos, estos algoritmos aíslan conceptos puros, eliminando la ambigüedad lingüística y reduciendo de nuevo el problema a una simple manipulación de proyecciones sobre subespacios de baja dimensión.

Marco pedagógico y recursos educativos en el aprendizaje automático

Dada la indiscutible posición que ocupa el álgebra lineal en la disciplina del machine learning, la formación de profesionales competentes ha impulsado la creación de trayectorias educativas y recursos orientados a cerrar la brecha entre la matemática pura y su implementación en código. Tradicionalmente, los libros de texto de aprendizaje automático asumen que el lector ya posee un dominio avanzado de las matemáticas y la estadística, dedicando a lo sumo un par de capítulos introductorios o apéndices a estos temas (Cavani, 2025). Sin embargo, la amplia adopción de estas tecnologías ha revelado una brecha de habilidades entre estudiantes y profesionales que no provienen de formaciones puramente matemáticas o físicas.

Para atender esta necesidad, han surgido recursos de acceso abierto y literatura especializada de alto valor. Un texto de referencia medular en esta transición es el libro Mathematics for Machine Learning de Marc Peter Deisenroth, A. Aldo Faisal y Cheng Soon Ong, que actúa como un puente

unificado que presenta el álgebra lineal, la geometría analítica y las descomposiciones matriciales con un enfoque directo en sus aplicaciones prácticas en algoritmos de machine learning. Este tipo de literatura se complementa con recursos visuales e intuitivos de gran relevancia en el ecosistema digital contemporáneo, como la serie de videos Essence of Linear Algebra del canal educativo 3Blue1Brown, frecuentemente recomendada por científicos de datos para construir una comprensión geométrica sólida antes de abordar el formalismo algebraico abstracto.

En la literatura clásica sobre la adquisición de fundamentos matemáticos rigurosos, las obras del profesor Gilbert Strang del Instituto Tecnológico de Massachusetts (MIT) continúan posicionándose como el estándar absoluto de la disciplina, ofreciendo una sólida comprensión de las factorizaciones matriciales y del análisis numérico necesario para sustentar desarrollos algorítmicos complejos. En combinación con cursos impartidos en plataformas masivas en línea por entidades de referencia como Coursera o DeepLearning.AI, los estudiantes e investigadores de hoy disponen de un ecosistema que democratiza el acceso al conocimiento técnico avanzado.

La mayoría de los investigadores sugieren firmemente un enfoque de aprendizaje pragmático, en el que los estudiantes comienzan construyendo proyectos prácticos y, a medida que surjan problemas numéricos o de optimización en el camino, el aprendizaje matemático se convierte en una necesidad natural que facilita la resolución de estos desafíos en lugar de ser una barrera de entrada puramente teórica. Una práctica pedagógica de alto rendimiento consiste en implementar desde cero algoritmos clásicos, como la regresión lineal o el descenso de gradiente, utilizando bibliotecas matriciales básicas como NumPy, para internalizar el comportamiento exacto de los

sistemas sobredeterminados y las consecuencias de la dispersión de los datos.

El análisis exhaustivo de las intersecciones entre las matemáticas vectoriales y las ciencias de la computación demuestra que el álgebra lineal no es un mero accesorio metodológico para el desarrollo del machine learning, sino su motor operativo y existencial directo (Brito et al., 2025). Desde las operaciones aritméticas más elementales sobre escalares y vectores que sustentan las regresiones estadísticas clásicas, hasta las complejas descomposiciones de rango bajo, como la SVD, que permiten destilar estructuras latentes en masivos conjuntos de datos no estructurados, el álgebra lineal provee el vocabulario inequívoco y estricto que dota de inteligencia matemática a las máquinas.

Se observa que la comprensión profunda de estos conceptos no solo permite a los profesionales aplicar algoritmos preexistentes como cajas negras cerradas, sino que resulta indispensable para diagnosticar comportamientos anómalos durante el entrenamiento de modelos complejos. Fenómenos analíticos tan disruptivos como la inestabilidad de las soluciones debido a la multicolinealidad, la pérdida exponencial de información en el proceso de proyección o el desvanecimiento de los gradientes en redes neuronales no pueden comprenderse ni mitigarse eficazmente sin una sólida base en las propiedades espectrales de las matrices y las leyes de transformación de los espacios vectoriales.

A medida que el campo avanza hacia el procesamiento de volúmenes de datos aún más inconmensurables y arquitecturas de inteligencia artificial generativa cada vez más masivas, el desafío inminente radicará en el desarrollo de algoritmos de álgebra lineal numérica que sean aún más

estables, dispersos y eficientes en el uso de recursos energéticos y de memoria. En última instancia, el futuro de la inteligencia artificial continuará indisolublemente ligado a la capacidad de la comunidad científica para seguir innovando en los métodos numéricos que gobiernan el comportamiento de los vectores, las matrices y los tensores que dan forma a nuestro entendimiento digital del mundo

Capítulo 3

Algoritmos de optimización para el ajuste de parámetros e hiper parámetros en modelos de aprendizaje automático

El éxito de los modelos de aprendizaje automático modernos, desde las redes neuronales profundas hasta los sistemas de inteligencia artificial generativa, depende críticamente de la precisión con la que se configuran sus variables internas y externas. La optimización, en su sentido más amplio, constituye el núcleo matemático que permite transformar un conjunto desordenado de datos en un modelo con capacidad predictiva y una generalización robusta. Este informe técnico analiza de manera exhaustiva los algoritmos diseñados para la navegación en paisajes de pérdida complejos, diferenciando entre el ajuste de parámetros internos y la optimización de hiper parámetros externos, y evaluando las herramientas y estrategias que definen el estado del arte en la disciplina.

Fundamentos de la optimización y la dualidad entre parámetros e hiper parámetros

En el marco del aprendizaje automático, la optimización se define como el proceso de búsqueda del conjunto de valores que minimiza o maximiza una función objetivo, generalmente una función de pérdida que cuantifica el error

del modelo. La distinción entre parámetros e hiperparámetros es fundamental para comprender la arquitectura de este proceso. Los parámetros son variables internas que el modelo estima directamente a partir de los datos durante la fase de entrenamiento, como los pesos y los sesgos en una red neuronal o los coeficientes en un modelo de regresión (Mishra et al., 2025). Estos valores definen la habilidad específica del modelo para resolver un problema dado y se registran como parte del conocimiento adquirido.

Por el contrario, los hiper parámetros son configuraciones externas que no pueden estimarse a partir de los datos y deben ser especificados por el profesional antes de que comience el entrenamiento. Estos rigen el comportamiento del algoritmo de aprendizaje, determinando aspectos como la velocidad de ajuste (tasa de aprendizaje), la complejidad de la arquitectura (número de capas o nodos) y la capacidad de regularización para evitar el sobreajuste. La optimización de hiper parámetros (HPO), también conocida como ajuste de hiper parámetros, busca identificar la configuración que maximiza el rendimiento del modelo en datos no vistos, a menudo mediante técnicas de validación cruzada para asegurar la estabilidad en el mundo real

La complejidad de la optimización en el aprendizaje profundo radica en la naturaleza de la superficie de pérdida, que suele ser altamente no convexa, con múltiples mínimos locales, llanuras extensas y puntos de silla en los que el gradiente es cercano a cero. La superación de estos obstáculos requiere algoritmos que no solo busquen el descenso más rápido, sino que también incorporen mecanismos de inercia, adaptación y exploración global.

Algoritmos de optimización de primer orden

para el ajuste de parámetros

Los algoritmos de primer orden utilizan la información de la derivada de la función de pérdida para actualizar los parámetros en la dirección opuesta al gradiente, buscando el punto de mínima energía. El desarrollo de estos métodos ha evolucionado desde el descenso de gradiente clásico hasta técnicas adaptativas sofisticadas que gestionan la incertidumbre y el ruido en los datos.

Descenso de gradiente estocástico y la introducción del momentum

El descenso de gradiente estocástico (SGD) es el pilar fundamental de la optimización en redes neuronales. A diferencia del descenso de gradiente por lotes (Batch Gradient Descent), que calcula el gradiente sobre todo el conjunto de datos, el SGD realiza actualizaciones utilizando una sola muestra o un pequeño lote (mini-lote) de datos. Esta naturaleza estocástica permite que el proceso sea computacionalmente eficiente y capaz de manejar conjuntos de datos masivos, además de introducir una fluctuación que puede ayudar al modelo a saltar fuera de mínimos locales deficientes (Tian et al., 2023).

Sin embargo, el SGD tradicional enfrenta desafíos significativos en regiones de la superficie de pérdida conocidas como cañones, donde la curvatura es mucho más pronunciada en una dirección que en otra, lo que provoca oscilaciones ineficientes. Para mitigar esto, se introdujo el concepto de Momentum, que incorpora un término de velocidad para acumular los gradientes de los pasos anteriores. La actualización se define

matemáticamente mediante la inclusión de un coeficiente de fricción γ que suaviza la trayectoria:

$$v_t = \gamma v_{t-1} + \eta \nabla_\theta J(\theta)$$

$$\theta = \theta - v_t$$

Esta acumulación de inercia permite al optimizador mantener una dirección constante y acelerar la convergencia en valles estrechos, lo que reduce el ruido de las actualizaciones estocásticas.

Paradigmas adaptativos: AdaGrad, RMSProp y Adam

La sensibilidad del rendimiento a la tasa de aprendizaje η llevó al desarrollo de algoritmos que ajustan dinámicamente este valor para cada parámetro individual. AdaGrad (Adaptive Gradient Algorithm) fue pionero en este enfoque al escalar la tasa de aprendizaje inversamente proporcional a la raíz cuadrada de la suma de los cuadrados de los gradientes históricos. Aunque es excelente para tratar datos dispersos y características infrecuentes, AdaGrad presenta una disminución monótona de la tasa de aprendizaje, lo que puede detener el entrenamiento prematuramente.

Para corregir la agresividad de AdaGrad, RMSProp utiliza un promedio móvil exponencial de los gradientes al cuadrado, lo que permite que el algoritmo olvide los gradientes muy antiguos y mantenga una tasa de aprendizaje efectiva durante todo el proceso. Esta innovación fue la base de Adam (Adaptive Moment Estimation), que combina las ventajas de Momentum y RMSProp al actualizar tanto el primer momento (media de los gradientes)

como el segundo momento (varianza no centrada).

AdaGrad ha demostrado ser extremadamente robusto y eficiente en una amplia variedad de tareas de aprendizaje profundo, convirtiéndose en el optimizador por defecto en muchos marcos de trabajo. No obstante, se ha observado que, en ciertos contextos, puede presentar una falta de generalización en comparación con el SGD con momentum bien sintonizado y es susceptible de fluctuaciones extremas al enfrentarse a datos ruidosos (véase la Tabla 9).

Tabla 9: Algoritmo según tareas de aprendizaje profundo

Algoritmo	**Mecanismo de actualización**	**Ventaja competitiva**	**Limitación principal**
SGD	Gradiente puro por muestra	Simplicidad y eficiencia en grandes datos	Convergencia lenta y oscilatoria
Momentum	Acumulación de velocidad	Acelera en valles y reduce ruido	Requiere ajuste manual de γ
AdaGrad	Suma histórica de gradientes cuadrados	Ideal para características dispersas	Desvanecimiento de la tasa de aprendizaje

RMSProp	Promedio móvil de gradientes cuadrados	Estabiliza la tasa en etapas tardías	Puede requerir ajuste de hiperparámetros
Adam	Media y varianza de gradientes	Convergencia rápida y robustez	Posible sobreajuste en mínimos agudos

Evolución hacia la estabilidad: AdamW, Yogi y Lion

La investigación contemporánea ha buscado refinar los algoritmos adaptativos para mejorar su capacidad de generalización y su eficiencia en el uso de la memoria. AdamW surge como una corrección necesaria al desacoplar el decaimiento de los pesos (weight decay) de la actualización del gradiente adaptativo, asegurando que la regularización se aplique de manera uniforme independientemente de la magnitud del gradiente (Pinto, 2023).

Por otro lado, Yogi introduce un control más estricto sobre el aumento de la tasa de aprendizaje efectiva, lo que previene que el optimizador realice pasos excesivamente grandes ante gradientes ruidosos y mejora la estabilidad en problemas no convexos complejos. Yogi se distingue por su capacidad para manejar superficies de pérdida rugosas, en las que Adam podría fallar debido a una adaptación excesivamente agresiva.

Una de las innovaciones más notables es Lion (EvoLved Sign Momentum), un optimizador descubierto mediante algoritmos genéticos por Google Brain. Lion utiliza únicamente el signo del gradiente acumulado, lo que

lo hace más eficiente en términos de memoria al no requerir almacenar los cuadrados de los gradientes. Se ha observado que Lion supera a Adam en el entrenamiento de Vision Transformers (ViT) y de modelos de difusión, logrando una mayor precisión con menos cómputo, aunque requiere una tasa de aprendizaje significativamente menor (3-10 veces menor que la de AdamW) para mantener la estabilidad.

Estrategias avanzadas para la optimización de hiperparámetros (HPO)

A diferencia de los parámetros, los hiperparámetros definen el espacio en el que el modelo aprende. El ajuste de estos valores es un proceso intensivo que requiere equilibrar la exploración de nuevas configuraciones con la explotación de regiones del espacio de búsqueda que han mostrado buenos resultados

Métodos de búsqueda no informados: rejilla y aleatoriedad

La búsqueda en Cuadrícula (Grid Search) es el enfoque más tradicional y exhaustivo. Consiste en definir manualmente un subconjunto de valores para cada hiperparámetro y evaluar todas las combinaciones posibles. Si bien garantiza encontrar el mejor punto dentro de la red definida, Grid Search es ineficiente en espacios de alta dimensión, ya que el número de modelos a entrenar crece exponencialmente con cada nuevo hiperparámetro, un fenómeno conocido como la maldición de la dimensionalidad

La Búsqueda Aleatoria (Random Search) reemplaza la enumeración exhaustiva por una selección estocástica basada en distribuciones de

probabilidad. Sorprendentemente, la investigación ha demostrado que la búsqueda aleatoria a menudo es más eficiente que la búsqueda en cuadrícula, especialmente cuando solo un pequeño número de hiperparámetros afecta realmente el rendimiento del modelo. Esto se debe a que Random Search explora un mayor número de valores únicos en cada dimensión, lo que aumenta las posibilidades de localizar los picos de rendimiento en el espacio de búsqueda.

Optimización bayesiana y modelado de sustitutos

La optimización bayesiana representa un salto cualitativo al tratar el ajuste de hiperparámetros como un problema de optimización de una función de caja negra costosa de evaluar. Este método construye un modelo probabilístico (sustituto) de la función de rendimiento a partir de los resultados de iteraciones previas (Huang et al., 2025). Existen dos enfoques principales para el modelado sustituto en este contexto:

1. **Procesos gaussianos (GP):** Estos modelos asumen que la función de rendimiento sigue una distribución normal multivariada. Proporcionan no solo una predicción de la métrica objetivo, sino también una cuantificación de la incertidumbre en cada punto del espacio. Aunque son potentes en espacios continuos, su costo computacional escala cúbicamente con el número de evaluaciones, lo que los hace menos adecuados para búsquedas muy largas o para espacios de alta dimensionalidad.
2. **Estimador de Parzen Estructurado en Árbol (TPE):** TPE es una técnica no paramétrica que modela las distribuciones de probabilidad de los hiperparámetros que conducen a buenos resultados ($l(x)$) y malos

resultados ($g(x)$) de forma separada. Al seleccionar puntos que maximizan la relación entre estas densidades, TPE puede manejar de forma eficiente hiperparámetros categóricos, discretos y condicionales (como el número de neuronas en una capa, que solo existe si la red tiene cierta profundidad).

La Optimización Bayesiana utiliza funciones de adquisición, como la Mejora Esperada (Expected Improvement), para decidir cuál es la siguiente configuración más prometedora para evaluar, logrando a menudo resultados superiores en una fracción del tiempo requerido por los métodos no bayesianos (Sun et al., 2025).

Comparativa de eficiencia en la sintonización de modelos

En estudios empíricos que comparan estos métodos, se observa una diferencia drástica en la velocidad de convergencia. En un experimento de optimización de un modelo Random Forest, se registraron los siguientes resultados (véase la Tabla 10):

Tabla 10: Comparativa de eficiencia en la sintonización de modelos Random forest

Método de HPO	**Total de ensayos**	**Iteración óptima**	**Puntuación (F1-score)**	**Tiempo de ejecución**
Grid Search	810	680	Máxima	Muy alto

Random Search	100	36	Mínima	El más bajo
Bayesiana (TPE)	100	67	Máxima	Medio

Estos datos subrayan que, si bien la búsqueda aleatoria es rápida, puede pasar por alto la configuración óptima, mientras que la optimización bayesiana alcanza la misma precisión que la búsqueda exhaustiva, pero con un ahorro de recursos de casi el 90% en términos de número de iteraciones.

Algoritmos metaheurísticos y bioinspirados en la optimización de parámetros

Cuando los paisajes de pérdida son extremadamente irregulares o no se dispone de información analítica sobre el gradiente, los algoritmos metaheurísticos ofrecen una alternativa poderosa basada en la exploración estocástica y en analogías con procesos naturales.

Algoritmos genéticos y evolutivos

Los algoritmos genéticos (GA) se basan en los principios de la evolución biológica. Mantienen una población de soluciones candidatas que evolucionan mediante mecanismos de selección de los más aptos, cruce (recombinación de rasgos de dos padres) y mutación (alteraciones aleatorias para mantener la

diversidad). En el aprendizaje automático, los GA son particularmente útiles para la búsqueda de arquitecturas neuronales (NAS) y para optimizar redes en las que las funciones de activación no son derivables (Sayin et al., 2025).

La principal ventaja de los GA es su capacidad intrínseca para escapar de los mínimos locales y explorar regiones del espacio de búsqueda. No obstante, evaluar la aptitud de cada individuo en una población grande puede resultar prohibitivamente caro, por lo que se requieren técnicas de paralelización y, a veces, el uso de modelos sustitutos para acelerar la evaluación.

Optimización por enjambre de partículas (PSO)

Inspirado en el comportamiento social de las bandadas de aves y los bancos de peces, el algoritmo PSO utiliza un conjunto de partículas que se mueven a través del espacio de búsqueda. Cada partícula ajusta su trayectoria basándose en su propia mejor posición conocida y en la mejor posición alcanzada por cualquier otro miembro del enjambre. PSO ha demostrado ser eficaz en la optimización de los pesos de redes neuronales recurrentes, logrando precisiones del 94 al 97% en tareas de clasificación complejas, a menudo superando a los métodos basados en el gradiente en términos de robustez frente a mínimos locales.

Recocido simulado y optimización inspirada en plantas

El Recocido Simulado (Simulated Annealing) es una técnica inspirada en la metalurgia, en la que un material se calienta y se enfría lentamente para alcanzar una estructura cristalina de mínima energía. En la optimización, el algoritmo acepta soluciones peores con una probabilidad que disminuye con

el tiempo (enfriamiento), lo que le permite saltar fuera de las trampas locales en las etapas iniciales de la búsqueda. Aunque es más lento que el descenso de gradiente en funciones continuas, resulta extremadamente versátil para problemas de optimización combinatoria y de ajuste de hiperparámetros en espacios discretos.

Una frontera emergente en este campo es la optimización inspirada en las plantas. Algoritmos como el Crecimiento de Rizomas o el Crecimiento Fototrópico han demostrado superar estadísticamente a los algoritmos inspirados en animales en el 97% de las funciones de referencia en espacios de alta dimensión. Estas técnicas aprovechan la capacidad de las plantas para distribuir recursos de manera descentralizada y resiliente, ofreciendo una nueva perspectiva sobre cómo gestionar la exploración en paisajes de pérdida extremadamente complejos.

Herramientas y ecosistemas de software para la optimización masiva

La implementación práctica de estos algoritmos se facilita mediante una serie de bibliotecas de código abierto y plataformas corporativas que automatizan el proceso de experimentación y seguimiento.

Frameworks líderes: Optuna, Hyperopt y Ray Tune

- **Optuna:** Actualmente es uno de los marcos más populares debido a su API define-by-run, que permite a los usuarios definir el espacio de búsqueda de forma imperativa mediante código Python estándar· Optuna integra algoritmos de vanguardia como TPE y CMA-ES, y cuenta con un

sistema de poda (pruning) que detiene automáticamente los experimentos que no muestran potencial, optimizando el uso de GPU/CPU.

- **Hyperopt:** Una de las bibliotecas más consolidadas, centrada en la optimización bayesiana distribuida es especialmente robusta para su uso con clústeres de Apache Spark o MongoDB, lo que permite escalar la búsqueda de hiper parámetros a cientos de nodos de manera asíncrona
- **Ray Tune:** Diseñado específicamente para la escalabilidad, permite ejecutar barridos de hiper parámetros distribuidos con menos de 10 líneas de código. Es compatible con estrategias avanzadas como el Entrenamiento Basado en Población (PBT) y con algoritmos de detección temprana como Hyperband y ASHA, y constituye una herramienta fundamental en entornos de entrenamiento de modelos a gran escala.

Plataformas de MLOps y gestión de experimentos

En el ámbito empresarial, herramientas como Weights & Biases (W&B) y MLflow proporcionan la infraestructura necesaria para registrar cada ejecución, visualizar las correlaciones entre hiper parámetros y métricas de salida, y asegurar la reproducibilidad de los modelos. Estas plataformas actúan como sistemas de registro que conectan el desarrollo del modelo con su despliegue y monitoreo en producción (véase la Tabla 11).

Tabla 11: Plataformas de MLOps y gestión de experimentos en el ámbito empresarial

Plataforma	Función principal	Soporte de optimización	Compatibilidad
W&B Sweeps	Experimentación y visualización	Bayesiana y Grid integrada	PyTorch, TF, Keras, etc.
MLflow	Ciclo de vida del modelo	Requiere librerías externas (Optuna)	Universal (REST API)
Amazon SageMaker	AutoML y entrenamiento en nube	Optimización bayesiana nativa	Ecosistema AWS
DataRobot	Automatización empresarial	HPO y selección de modelos de auto.	Datos estructurados/LLM
Google Vertex AI	NAS e infraestructura cloud	Búsqueda de arquitectura y HPO	Google Cloud Platform

Desafíos técnicos en la optimización de

modelos de alta dimensión

A medida que los modelos transitan hacia escalas de billones de parámetros, los desafíos de optimización se vuelven más agudos, lo que exige innovaciones en eficiencia de cómputo y en estabilidad numérica.

La maldición de la dimensionalidad y los puntos de silla

En espacios de parámetros e hiperparámetros de alta dimensión, el volumen del espacio crece tan rápido que los datos se vuelven dispersos, lo que dificulta la identificación de regiones óptimas. Además, en redes neuronales profundas, la presencia de puntos de silla es mucho más común que la de mínimos locales (Barry et al., 2021). Los optimizadores modernos deben ser capaces de navegar por estas regiones donde el gradiente es cero pero el punto no es un mínimo, utilizando información de segundo orden (Hessiana) o mecanismos de inercia que proporcionan el impulso necesario para cruzar estas llanuras.

Robustez frente al ruido y generalización

Un problema persistente en la optimización es el compromiso entre la precisión en el entrenamiento y la capacidad de generalización. Algoritmos como Adam tienden a converger hacia mínimos agudos, que pueden ser muy precisos en los datos de entrenamiento, pero fallan ante pequeñas variaciones en los datos de prueba. En contraste, el SGD suele encontrar mínimos planos, asociados con una mayor robustez y una mayor capacidad de generalización. Optimizadores como Yogi y Lion han sido diseñados para mitigar este efecto, buscando soluciones que mantengan la estabilidad incluso ante distribuciones de datos cambiantes.

Sostenibilidad y eficiencia energética (LLMOps)

En la era de los Modelos de Lenguaje de Gran Tamaño (LLM), la eficiencia del ajuste de hiper parámetros no es solo una cuestión de rendimiento, sino de sostenibilidad. Las operaciones de LLMOps ponen especial énfasis en minimizar los requisitos de potencia de cómputo. El uso de algoritmos de multi-fidelidad, que permiten evaluar modelos en subconjuntos de datos o con menos épocas antes de comprometer recursos masivos, se ha vuelto una práctica estándar para reducir la huella de carbono y los costos operativos de la IA.

Síntesis de hallazgos y perspectivas de futuro en optimización

La trayectoria de los algoritmos de optimización refleja una transición de métodos manuales y heurísticos a sistemas automatizados e inteligentes que aprenden a optimizar.

La distinción entre parámetros e hiper parámetros sigue siendo el eje en torno al cual se basa el entrenamiento de modelos. Mientras que los parámetros se ajustan mediante gradientes cada vez más sofisticados que incorporan momentos y adaptación (como AdamW y Lion), los hiper parámetros requieren estrategias de nivel superior donde la optimización bayesiana y los algoritmos evolutivos han demostrado ser los más eficaces para navegar espacios complejos sin requerir un coste computacional inasumible (Blume et al., 2021).

De cara al futuro, la integración de la inteligencia artificial en el propio

proceso de diseño de algoritmos (como lo demuestra el descubrimiento de Lion) sugiere que los optimizadores del mañana serán cada vez más específicos para cada dominio, capaces de adaptarse dinámicamente a la arquitectura del modelo y a la naturaleza de los datos. Asimismo, el surgimiento de la computación cuántica y los algoritmos inspirados en la mecánica cuántica prometen abrir nuevas vías para resolver problemas de optimización no convexos que actualmente están fuera del alcance de la computación clásica.

En síntesis, la elección del algoritmo de optimización no debe considerarse una decisión trivial, sino estratégica, que incide directamente en la precisión, la capacidad de generalización y la viabilidad económica de cualquier proyecto de aprendizaje automático. La adopción de marcos de trabajo como Optuna o Ray Tune, junto con una comprensión profunda de las dinámicas de convergencia de optimizadores como Adam, SGD o L-BFGS, constituye la base necesaria para cualquier profesional que aspire a construir sistemas de inteligencia artificial de alto rendimiento en el panorama tecnológico actual y futuro.

Capítulo 4

Probabilidad y Estadística para Machine Learning: Marco Integral para el Análisis y la Toma de Decisiones

La evolución contemporánea de la inteligencia artificial ha desplazado el paradigma de la computación determinista hacia un modelo basado en la gestión sistemática de la incertidumbre. En este contexto, la probabilidad y la estadística no constituyen meras herramientas auxiliares, sino la infraestructura ontológica sobre la que se erige el machine learning. La capacidad de un algoritmo para aprender de los datos, generalizar patrones y facilitar la toma de decisiones complejas en entornos ruidosos depende intrínsecamente de su capacidad para cuantificar lo desconocido.

Este capítulo analiza en profundidad cómo las teorías matemáticas de la probabilidad y los métodos de inferencia estadística convergen para transformar datos en bruto en inteligencia accionable, explorando desde los fundamentos axiomáticos hasta las aplicaciones avanzadas en sectores estratégicos como las finanzas, la genética y los sistemas autónomos.

El imperativo probabilístico en el aprendizaje automático

El modelado mediante machine learning surge precisamente como

respuesta a la imposibilidad de establecer correlaciones perfectas y deterministas en el mundo real. Los datos recolectados suelen ser incompletos, ruidosos y sujetos a una variabilidad inherente que los modelos matemáticos tradicionales no pueden captar. Por tanto, la disciplina ha adoptado un enfoque probabilístico que permite modelar no solo certezas, sino también distribuciones de probabilidad. Esta perspectiva es crítica para cuantificar la confianza de las predicciones y medir la seguridad con la que un modelo estadístico opera en condiciones de incertidumbre

Definiciones fundamentales y espacios de eventos

Para comprender la mecánica del aprendizaje estadístico, es imperativo establecer una base terminológica rigurosa. Una variable aleatoria se define como la representación numérica de un fenómeno aleatorio, cuyos valores posibles son los resultados de un proceso no determinista (García et al., 2013). Estas se clasifican en dos categorías esenciales: discretas y continuas. Las variables discretas asumen un número finito o infinito de valores, como el número de correos electrónicos no deseados recibidos en un intervalo de tiempo o el resultado binario de un diagnóstico médico. Por el contrario, las variables continuas operan en un espectro infinito dentro de un rango determinado, capturando magnitudes como la altura de un individuo, la temperatura ambiental o la inversión financiera en un mercado volátil

El concepto de evento, definido como el conjunto de uno o más resultados de un proceso aleatorio, permite estructurar el espacio muestral sobre el cual operan los algoritmos La probabilidad de un evento A, denotada como $P(A)$, es una medida numérica de su verosimilitud que oscila estrictamente entre 0 y 1 En el machine learning, esta noción se extiende

frecuentemente a la probabilidad condicional $P(A|B)$, que evalúa la probabilidad de que ocurra el evento A dado que el evento B ya ha tenido lugar. Esta relación es el motor de los modelos de regresión logística y de clasificación, en los que se busca predecir un resultado (como el abandono de un cliente o el riesgo de impago) a partir de características específicas observadas en los datos de entrenamiento (García et al., 2013) (véase la Tabla 12).

Tabla 12: Ejemplos en machine learning según el tipo de variable

Tipo de Variable	**Características**	**Ejemplos en Machine Learning**
Discreta	Valores contables y finitos.	Clases de clasificación, recuento de eventos (Poisson).
Continua	Valores en un rango infinito.	Pesos de redes neuronales, predicción de ingresos.
Determinista	Resultado previsible sin error.	Programación clásica con reglas fijas.

Probabilística	Incorpora el factor de azar.	Modelos de regresión, procesos estocásticos.

El paradigma bayesiano: inferencia y actualización de creencias

El teorema de Bayes es una de las proposiciones más influyentes del cálculo de probabilidades y constituye el corazón de la inferencia bayesiana en machine learning. Este teorema permite actualizar sistemáticamente las creencias iniciales ante la llegada de nueva información empírica, lo que se denomina invertir las probabilidades condicionales (Sun et al., 2025).

Mecánica del Teorema de Bayes

La fórmula fundamental del teorema establece que la probabilidad de un evento A dado un evento B es igual a la probabilidad de B dado A, multiplicada por la probabilidad inicial de A y dividida por la probabilidad total de B:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

En el contexto del aprendizaje estadístico, los términos adquieren una importancia operativa específica:

1. **Probabilidad a priori ($P(A)$):** Representa el conocimiento inicial o la frecuencia marginal de un evento antes de considerar la evidencia actual

2. **Verosimilitud ($P(B|A)$):** Es la probabilidad de observar la evidencia B si la hipótesis A fuera cierta
3. **Probabilidad a posteriori ($P(A|B)$):** Es el resultado final del análisis, la probabilidad actualizada de la hipótesis tras observar los datos
4. **Evidencia ($P(B)$):** El factor de normalización que asegura que las probabilidades sumen la unidad

Este enfoque se distingue por su naturaleza secuencial: la información adicional adquirida modifica la probabilidad inicial, lo que permite que los modelos se adapten dinámicamente. Aplicaciones prácticas de esta lógica incluyen el filtrado de correo basura, en el que la aparición de ciertas palabras clave (evidencia) modifica la probabilidad a priori de que un mensaje sea spam, y el diagnóstico médico, en el que un resultado positivo en una prueba actualiza la probabilidad a priori de padecer una enfermedad.

El clasificador Naïve Bayes

Una de las implementaciones más extendidas de esta teoría es el clasificador Naïve Bayes (NBC), un modelo probabilístico que asume la independencia condicional entre los atributos de entrada dada la clase. Aunque esta suposición de independencia suele ser una simplificación excesiva de la realidad —de ahí el término ingenuo o *naïve*—, el modelo es extremadamente robusto, rápido de entrenar y eficaz en problemas de alta dimensionalidad. La probabilidad posterior se obtiene mediante el producto de las probabilidades condicionales individuales de cada atributo, lo que reduce el crecimiento de parámetros de una escala exponencial a una lineal

Inferencia estadística y validación de modelos

Mientras que la probabilidad se ocupa de predecir resultados a partir de parámetros conocidos, la estadística inferencial recorre el camino inverso: utiliza muestras de datos para inferir las propiedades de una población completa. En la ciencia de datos, este proceso es esencial para validar si los patrones descubiertos por un algoritmo son generalizables o si son simplemente producto del azar.

El contraste de hipótesis y el valor p

El rigor científico en la validación de modelos se basa en las pruebas de hipótesis. El proceso se inicia con la formulación de la hipótesis nula (H_0), que postula que no existe un efecto o diferencia significativa entre los grupos analizados, y la hipótesis alternativa (H_1), que sugiere lo contrario.

La evaluación de estas hipótesis depende del cálculo del valor p (*p-value*), que representa la probabilidad de observar los datos obtenidos (o unos más extremos) asumiendo que H_0 es verdadera Si el valor p es inferior al nivel de significancia α (típicamente 0.05), se rechaza la hipótesis nula, concluyendo que existe evidencia estadística para aceptar la intervención o el impacto medido Este marco es fundamental para evaluar si, por ejemplo, una nueva versión de un algoritmo de recomendación realmente mejora las ventas en comparación con la versión anterior.

Tipos de pruebas y errores de decisión

La elección de la prueba estadística adecuada depende del diseño

experimental y de la naturaleza de los datos:

- **Pruebas t de Student:** Se emplean para comparar las medias de dos grupos. Pueden ser de una muestra, de dos muestras independientes o pareadas (antes y después de un tratamiento)
- **ANOVA (Análisis de Varianza):** Extensión de la prueba t para comparar más de dos grupos simultáneamente
- **Prueba Chi-cuadrado (χ^2):** Aplicada a variables categóricas para determinar la asociación entre ellas, como la relación entre el género y la preferencia de producto

Al realizar estas pruebas, es inevitable considerar la posibilidad de errores. El error tipo I ocurre cuando se rechaza una hipótesis verdadera (falso positivo), mientras que el error tipo II ocurre cuando se acepta una hipótesis falsa (falso negativo). La potencia de una prueba estadística es la probabilidad de detectar una diferencia verdadera y está influida directamente por el tamaño de la muestra y la magnitud del efecto buscado.

Intervalos de confianza y estimación de parámetros

La estadística inferencial no solo busca rechazar hipótesis, sino también estimar parámetros poblacionales mediante intervalos de confianza (IC). Un IC del 95% indica que, si se repitiera el muestreo múltiples veces, el 95% de los intervalos calculados contendrían el valor poblacional. Estos intervalos son herramientas de transparencia cruciales, ya que comunican la precisión de las estimaciones y permiten tomar decisiones informadas sobre la confiabilidad de un modelo predictivo.

Análisis exploratorio de datos (EDA): la fase

de descubrimiento

Ningún modelo de machine learning puede ser efectivo si se ignora la calidad y la estructura de los datos de entrada. El Análisis Exploratorio de Datos (EDA) es un proceso iterativo y visual diseñado para comprender la estructura de los datos, detectar anomalías y refinar las hipótesis iniciales antes del modelado formal.

Tratamiento de datos ausentes y atípicos

Durante la etapa de limpieza de datos, el analista debe identificar y gestionar las inconsistencias. Los datos ausentes representan un desafío significativo para las técnicas de aprendizaje automático y pueden introducir sesgos o errores de ejecución. Las estrategias comunes incluyen la eliminación de registros incompletos (útil solo en grandes volúmenes de datos), la imputación mediante la media o la mediana, o el uso de modelos predictivos para estimar los valores faltantes a partir de otras variables.

La detección de datos atípicos (*outliers*) es igualmente crítica. Los valores que difieren significativamente del resto pueden distorsionar los cálculos de varianza y reducir la potencia de los análisis estadísticos. Sin embargo, su tratamiento debe ser cuidadoso: mientras que algunos son errores de recolección que deben eliminarse, otros pueden ser indicadores de eventos raros pero legítimos (como fraudes bancarios) que el modelo debe ser capaz de identificar.

Técnicas de visualización y resumen

El EDA se apoya extensamente en herramientas gráficas que permiten

discernir patrones invisibles en las tablas numéricas:

- **Histogramas y gráficos de densidad:** identifican la forma de la distribución y detectan comportamientos multimodales o sesgos.
- **Diagramas de caja (Boxplots):** resumen la dispersión de los datos mediante cuartiles y destacan visualmente los valores atípicos.
- **Gráficos de dispersión y matrices de correlación** revelan la fuerza y la dirección de las relaciones entre variables, lo que ayuda a seleccionar los predictores más relevantes y a evitar la redundancia.

Este proceso de exploración no solo prepara el terreno para el modelado, sino que a menudo conduce a hallazgos inesperados que cambian la dirección de la investigación.

Distribuciones de probabilidad clave en el aprendizaje estadístico

El comportamiento de los datos reales suele ajustarse a modelos teóricos conocidos como distribuciones de probabilidad. Identificar qué distribución sigue una variable es un paso previo necesario para aplicar muchos algoritmos de machine learning.

La Distribución Normal y el Teorema Central del Límite

La distribución normal, o gaussiana, es el pilar de la estadística paramétrica. Se caracteriza por ser simétrica respecto a su media y por estar definida por su desviación estándar. Su importancia radica en el Teorema Central del Límite, que establece que la suma de un gran número de variables independientes e idénticamente distribuidas tiende a seguir una distribución

normal, independientemente de la distribución original de dichas variables. Muchos algoritmos, como la regresión lineal y el análisis de componentes principales (PCA), asumen la normalidad de los datos o de los residuos para garantizar resultados óptimos

Distribuciones discretas: binomial y Poisson

En escenarios donde se trabaja con datos de recuento o eventos binarios, se recurre a otras distribuciones:

- **Distribución Binomial:** Modela el número de éxitos en una serie de n ensayos independientes, con una probabilidad fija de éxito p. Es la base para entender experimentos de Bernoulli y se aplica frecuentemente en la clasificación binaria
- **Distribución de Poisson:** utilizada para modelar la probabilidad de que ocurra un número determinado de eventos en un intervalo fijo de tiempo o de espacio, asumiendo una tasa de ocurrencia constante. Es ideal para modelar llegadas de clientes a un servicio, llamadas a un centro de atención o la aparición de defectos en una línea de producción

El conocimiento de estas distribuciones permite a los científicos de datos realizar simulaciones probabilísticas y optimizar procesos mediante la predicción de eventos futuros con base en frecuencias históricas.

Reducción de dimensionalidad: Análisis de Componentes Principales (PCA)

En el machine learning moderno, es común enfrentarse a conjuntos de

datos con cientos de variables, lo que incrementa la complejidad computacional y el riesgo de sobreajuste. El Análisis de Componentes Principales (PCA) es una técnica estadística multivariante diseñada para simplificar estos datos conservando la mayor cantidad de información posible.

Mecanismo matemático del PCA

El PCA transforma las variables originales, que suelen estar correlacionadas, en un nuevo conjunto de variables no correlacionadas llamadas componentes principales (PC). Este proceso implica varios pasos críticos:

1. **Normalización:** Es imperativo estandarizar los datos para que todas las variables tengan media 0 y desviación estándar 1, evitando que las variables con escalas mayores dominen el análisis.
2. **Matriz de Covarianza:** Se calcula para identificar las correlaciones entre las variables originales y comprender cómo varían conjuntamente.
3. **Vectores y Valores Propios:** Se obtienen a partir de la matriz de covarianza. Los vectores propios definen la dirección de los nuevos componentes (ejes), mientras que los valores propios indican la cantidad de varianza que captura cada componente.
4. **Selección de Componentes:** El primer componente principal (PC1) explica la mayor parte de la varianza total. El segundo componente (PC2) captura la siguiente mayor varianza y es estrictamente perpendicular al primero.

Aplicaciones estratégicas y de negocio

El PCA no es solo una técnica de preprocesamiento, sino también una

herramienta para el descubrimiento de estructuras ocultas. Sus aplicaciones incluyen:

- **Visualización de datos de alta dimensión:** proyectar datos complejos en gráficos 2D o 3D para facilitar su interpretación
- **Compresión de imágenes:** Reducir el tamaño de los archivos visuales conservando los detalles esenciales
- **Optimización de campañas publicitarias:** condensar múltiples métricas (clics, impresiones, conversiones) para identificar qué factores impulsan realmente el rendimiento del negocio
- **Detección de anomalías:** Al identificar el patrón general de los datos, las desviaciones significativas en el espacio de componentes principales se hacen evidentes.

Teoría de la decisión estadística y funciones de pérdida

La finalidad última de los modelos probabilísticos es facilitar la toma de decisiones óptimas en entornos inciertos. La teoría de la decisión proporciona un marco lógico para elegir entre diversas alternativas, basándose en principios de coherencia y de maximización de la utilidad.

Elementos de un problema de decisión

Un problema de decisión se define formalmente mediante cuatro elementos: el espacio de opciones (alternativas posibles), el espacio de eventos inciertos (lo que puede hacer la naturaleza), las consecuencias de cada par de opción-evento y la relación de preferencia entre dichas consecuencias. En este

ámbito, se suelen emplear diferentes estrategias de elección:

- **Criterio de Utilidad Esperada:** Se asigna una utilidad a cada consecuencia y se elige la opción que maximiza el promedio ponderado de las utilidades de los eventos.
- **Estrategia Minimax (Pesimista):** Se elige la opción cuya peor consecuencia posible sea la menos perjudicial. Es una estrategia de minimización del riesgo máximo.
- **Estrategia Optimista:** Se basa en la mejor consecuencia posible de cada opción.

Funciones de pérdida y minimización del riesgo

En machine learning, el proceso de entrenamiento de un modelo es, en esencia, un problema de minimización del riesgo empírico. La función de pérdida cuantifica qué tan incorrectas son las predicciones de un modelo comparándolas con las etiquetas reales

Existen diversas funciones de pérdida según el objetivo del modelo:

- **Error Cuadrático Medio (ECM/L2):** Penaliza drásticamente los errores grandes al elevar la diferencia al cuadrado. Es ideal cuando los errores atípicos se consideran importantes y deben ser evitados a toda costa
- **Error Absoluto Medio (MAE/L1):** Trata todos los errores con el mismo peso, independientemente de su magnitud. Es más robusto frente a valores atípicos que el ECM.
- **Pérdida de Huber:** Combina las ventajas de L1 y L2, siendo cuadrática para errores pequeños y lineal para los grandes, lo que reduce la sensibilidad a los valores atípicos extremos

- **Entropía cruzada:** Se utiliza en problemas de clasificación para medir la diferencia entre las distribuciones de probabilidad reales y las predichas.

El objetivo fundamental es encontrar los parámetros del modelo (pesos y sesgos) que minimicen la función de pérdida sobre los datos de entrenamiento, utilizando algoritmos de optimización como el descenso de gradiente.

Modelado avanzado: redes bayesianas y procesos gaussianos

Para abordar problemas complejos en los que las dependencias entre variables no son lineales ni directas, la inteligencia artificial utiliza modelos probabilísticos gráficos. Estos sistemas permiten razonar sobre la incertidumbre al combinar la teoría de grafos con la teoría de la probabilidad (Sun et al., 2025).

Redes Bayesianas y Razonamiento Causal

Una red bayesiana (RB) es una representación gráfica de dependencias en la que los nodos representan variables aleatorias y los arcos dirigidos representan relaciones de dependencia directa. Estas redes son potentes herramientas generativas y predictivas porque permiten:

- **Incorporar conocimiento experto:** Se pueden definir estructuras y probabilidades *a priori* basadas en el saber de profesionales del dominio (como genetistas o médicos)
- **Manejar datos incompletos:** Las RB pueden realizar inferencias incluso cuando faltan valores de algunas variables, utilizando la información de

los nodos adyacentes para estimar las probabilidades

- **Interpretabilidad:** A diferencia de las redes neuronales de caja negra, las RB ofrecen una estructura transparente que permite entender el porqué de una decisión, lo cual es crítico en finanzas y salud.

Aplicaciones prácticas de las RB incluyen el diagnóstico de fallos en procesadores (Intel), la predicción de errores en redes de software definidas por API, y la mejora genética vegetal para maximizar la calidad de los cultivos

Procesos gaussianos y cuantificación de la confianza

Los procesos gaussianos (GP) proporcionan un enfoque no paramétrico para la regresión. En lugar de predecir un único valor escalar, un GP ofrece una distribución de posibles funciones que pasan por los puntos de datos observados. Esto se traduce en una predicción acompañada de un intervalo de confianza claro: donde hay pocos datos, la incertidumbre aumenta, lo que alerta al decisor sobre la falta de fiabilidad del modelo en esa región del espacio. Se utilizan ampliamente en la robótica para adaptar trayectorias en entornos inciertos y en la previsión de series temporales financieras.

Estadística vs. Machine Learning: Convergencia y Diferencias Epistemológicas

A menudo se confunden los términos, pero su lógica inferencial y sus objetivos prácticos difieren profundamente. La estadística inferencial busca entender y explicar las relaciones causales a partir de supuestos teóricos estrictos y de muestras limitadas. Su métrica de éxito es la significación estadística y la interpretabilidad de los parámetros

Por el contrario, el machine learning prioriza la precisión predictiva y la generalización a nuevos datos, operando con frecuencia sin asumir una forma funcional específica y manejando volúmenes masivos de información. Mientras que un estadístico preguntaría: ¿Cómo afecta la variable X a la variable Y?, un ingeniero de ML preguntaría: ¿Qué tan bien puedo predecir Y a partir de X? En la práctica moderna, el científico de datos más eficaz es aquel que integra ambos enfoques: utiliza la estadística para validar hipótesis y limpiar datos, y el machine learning para escalar predicciones y automatizar procesos complejos

El ciclo de vida del proyecto de Machine Learning Estadístico

La implementación exitosa de estas técnicas en la empresa sigue un proceso estructurado:

1. **Definición del objetivo empresarial:** Identificar problemas que generen valor tangible y criterios de éxito medibles
2. **Preprocesamiento de datos:** Limpieza, normalización y transformación de datos en bruto para asegurar que sean pertinentes y estructurados
3. **Modelado y entrenamiento:** selección de algoritmos probabilísticos y ajuste de parámetros mediante la minimización de funciones de pérdida.
4. **Evaluación y optimización:** Uso de métricas avanzadas (Accuracy, Recall, Precision, F1-Score) para medir el rendimiento sobre datos no vistos y ajustar hiperparámetros
5. **Despliegue y mejora continua:** Los sistemas deben adaptarse y aprender a partir de los nuevos datos que procesan en tiempo real,

mejorando progresivamente su precisión.

La probabilidad y la estadística no son meros requisitos académicos para el estudio del machine learning; son las leyes fundamentales que rigen el comportamiento de los algoritmos en un universo incierto. Desde el Teorema de Bayes, que permite la actualización dinámica del conocimiento, hasta el PCA, que destila la esencia de los datos multidimensionales, cada técnica analizada contribuye a la creación de sistemas de IA más robustos, interpretables y precisos.

Para las organizaciones, la inversión en estas capacidades analíticas se traduce en mejoras directas en la toma de decisiones, la optimización de recursos y una comprensión más profunda del comportamiento del cliente. La transición de un enfoque reactivo basado en la intuición a un modelo proactivo basado en datos probabilísticos es hoy en día una ventaja competitiva insalvable. En última instancia, el dominio de la probabilidad y la estadística permite a los profesionales no solo predecir lo que ocurrirá, sino también cuantificar con qué grado de certeza ocurrirá, transformando el riesgo en una variable gestionable y estratégica

Capítulo 5
El Cálculo Multivariable como Eje Vertebrador del Entrenamiento de Redes Neuronales Profundas

La revolución del aprendizaje profundo ha transformado sectores enteros, desde la visión por computadora hasta el procesamiento del lenguaje natural, y se ha consolidado como el paradigma más exitoso en el campo del aprendizaje automático. Este éxito no es una coincidencia tecnológica aislada, sino el resultado de la aplicación rigurosa de principios matemáticos fundamentales.

En el núcleo de esta capacidad de aprendizaje reside el cálculo multivariable, una rama de las matemáticas que proporciona el marco teórico y las herramientas computacionales necesarias para optimizar funciones complejas en espacios de parámetros de dimensiones extremadamente altas. El entrenamiento de una red neuronal es, en su esencia más pura, un problema de optimización en el que el cálculo permite navegar por un paisaje de pérdida intrincado para encontrar configuraciones de pesos que minimicen el error.

Fundamentos Matemáticos del Aprendizaje Profundo

Para comprender la importancia del cálculo multivariable en la inteligencia artificial moderna, es imperativo analizar la tríada matemática que

sustenta el aprendizaje profundo: el álgebra lineal, la teoría de la probabilidad y el cálculo multivariable. El álgebra lineal ofrece un lenguaje para representar los datos y los parámetros del modelo mediante tensores, matrices y vectores. La teoría de la probabilidad permite cuantificar la incertidumbre, gestionar la inicialización aleatoria de los pesos y modelar la distribución de los datos de entrada. Sin embargo, es el cálculo multivariable el que insufla vida a estas estructuras estáticas, transformándolas en sistemas dinámicos capaces de aprender mediante el ajuste iterativo (Murad et al., 2021).

El aprendizaje profundo utiliza redes neuronales multicapa entrenadas con grandes conjuntos de datos para resolver tareas complejas de procesamiento de información Cada capa de la red realiza una transformación matemática de los datos recibidos, y el cálculo multivariable es el encargado de determinar cómo cada pequeño cambio en los parámetros de estas transformaciones afecta al rendimiento global del sistema Sin esta capacidad de análisis de sensibilidad, las redes neuronales serían incapaces de corregir sus errores de manera sistemática.

La Función de Pérdida y el Objetivo de Optimización

El proceso de entrenamiento comienza con la definición de una función de pérdida o coste (L), que sirve como indicador de la precisión del modelo. Esta función toma las predicciones de la red y las compara con las etiquetas reales o con los valores objetivos. En términos de cálculo, la función de pérdida es una función escalar de múltiples variables, donde las variables son los pesos (w) y los sesgos (b) de todas las capas de la red (véase la Tabla 13).

Tabla 13: Proceso de entrenamiento y precisión del algoritmo

Componente Matemático	**Descripción en el Contexto de Redes Neuronales**	**Impacto en el Entrenamiento**
Parámetros (θ)	Conjunto de todos los pesos y sesgos del modelo.	Define la capacidad de representación de la red.
Espacio de Parámetros	Espacio de alta dimensión en el que cada punto corresponde a una configuración de la red.	La topología de este espacio determina la facilidad de entrenamiento.
Función de Pérdida (L)	Superficie escalar que asigna parámetros a un error.	Proporciona la señal necesaria para ajustar los pesos.
Gradiente (∇L)	Vector de derivadas parciales con respecto a cada parámetro.	Indica la dirección de mayor crecimiento del error.

El objetivo fundamental del entrenamiento es encontrar el conjunto de parámetros θ^* que minimice la función de pérdida:

$$\theta^* = \arg\min_{\theta} L(\theta)$$

Para lograr esto en redes que contienen millones de parámetros, se emplea el descenso de gradiente, un método iterativo que utiliza la información derivada del cálculo multivariable para descender por la superficie de pérdida hasta alcanzar un mínimo local o global

El Descenso de Gradiente y la Geometría del Aprendizaje

El descenso de gradiente es el algoritmo de optimización más utilizado en el aprendizaje profundo. Su funcionamiento se basa en una propiedad fundamental del gradiente: en cualquier punto de una superficie diferenciable, el vector del gradiente apunta en la dirección del ascenso más pronunciado. Por lo tanto, para reducir el error, el algoritmo debe desplazar los parámetros en la dirección opuesta al gradiente (Zhu et al., 2022).

El Gradiente como Vector de Sensibilidades

En el contexto multivariable, el gradiente de la función de pérdida ∇L es un vector que contiene las derivadas parciales de L con respecto a cada uno de los parámetros del modelo:

$$\nabla L = \left[\frac{\partial L}{\partial \theta_1}, \frac{\partial L}{\partial \theta_2}, \ldots, \frac{\partial L}{\partial \theta_n}\right]^T$$

Cada componente $\frac{\partial L}{\partial \theta_i}$ representa la sensibilidad de la pérdida ante un

cambio infinitesimal en el parámetro θ_i. Si una derivada parcial es positiva, aumentar el parámetro aumentará la pérdida; si es negativa, disminuirlo disminuirá la pérdida. El cálculo multivariable permite que el modelo realice ajustes simultáneos en todas las dimensiones del espacio de parámetros, coordinando la actualización de millones de pesos para mejorar el rendimiento colectivo del sistema.

Regla de Actualización y Tasa de Aprendizaje

La actualización de los parámetros se realiza mediante la siguiente regla:

$$\theta_{t+1} = \theta_t - \eta \nabla L(\theta_t)$$

Aquí, η representa la tasa de aprendizaje, un hiper parámetro crítico que controla el tamaño del paso dado en la dirección del gradiente negativo. Si la tasa de aprendizaje es demasiado alta, el sistema puede sobrepasar el mínimo y divergir; si es demasiado baja, el entrenamiento será excesivamente lento y puede quedar atrapado en regiones de la superficie de pérdida con pendientes insignificantes. La determinación de la tasa de aprendizaje óptima está íntimamente ligada a la curvatura de la función de pérdida, la cual se analiza mediante derivadas de segundo orden organizadas en la matriz hessiana.

Retropropagación: La Regla de la Cadena en Redes Profundas

La retropropagación (backpropagation) es el algoritmo que permite

calcular de manera eficiente el gradiente de la función de pérdida con respecto a todos los pesos de una red neuronal multicapa. Aunque a menudo se presenta como un concepto complejo, la retropropagación es, fundamentalmente, la aplicación sistemática de la regla de la cadena del cálculo multivariable a un grafo de computación (Kim et al., 2021).

De la Regla de la Cadena Univariable a la Multivariable

En el cálculo básico, si $y = f(g(x))$, la derivada de y con respecto a x se obtiene multiplicando las derivadas de las funciones componentes: $\frac{dy}{dx} = \frac{dy}{dg} \cdot \frac{dg}{dx}$. Sin embargo, en una red neuronal, la salida de una neurona de la capa l suele influir en múltiples neuronas de la capa $l + 1$. Esto requiere la versión multivariable de la regla de la cadena, donde las contribuciones al error de cada camino se suman

Si una función de pérdida L depende de variables intermedias $u_1, u_2, \ldots, u_m$, y cada una de estas variables depende a su vez de un peso w, la derivada total de L con respecto a w es la suma de los productos de las derivadas parciales a lo largo de todos los caminos posibles:

$$\frac{\partial L}{\partial w} = \sum_{i-1}^{m} \frac{\partial L}{\partial u_i} \frac{\partial u_i}{\partial w}$$

Esta suma refleja cómo se distribuye el error a lo largo de la arquitectura de la red. La retropropagación aprovecha esta estructura procesando los gradientes desde la capa de salida hacia la capa de entrada,

reutilizando cálculos intermedios para evitar la explosión combinatoria que ocurriría si se calculara cada derivada de forma independiente

El Papel de las Funciones de Activación

El cálculo de los gradientes depende intrínsecamente de la diferenciabilidad de las funciones de activación. Estas funciones introducen la no linealidad necesaria para que la red aprenda patrones complejos. En el proceso de retropropagación, la derivada de la función de activación actúa como un modulador del gradiente que fluye a través de la neurona (Akter y Haider, 2025)

Por ejemplo, al utilizar la función sigmoide $\sigma(z) = \frac{1}{1+e^{-z}}$, su derivada tiene la propiedad conveniente de poder expresarse en términos de la propia salida$\sigma'(z) = \sigma(z)(1 - \sigma(z))$. Esta propiedad simplifica enormemente el cálculo del gradiente, permitiendo actualizaciones rápidas de los pesos basándose únicamente en los valores de activación ya calculados durante el paso hacia adelante (véase la Tabla 14).

Tabla 14: Función de activación y el impacto en el flujo de gradiente

Función de Activación	**Definición Matemática**	**Derivada (∂z∂a)**	**Impacto en el flujo de gradiente**
Sigmoide	$\frac{1}{1+e^{-z}}$	$a(1 - a)$	Tiende a saturarse, lo que provoca el desvanecimiento del

			gradiente.
Tanh	$\frac{e^z - e^{-z}}{e^z + e^{-z}}$	$1 - a^2$	Centrada en cero, con una convergencia superior a la de la sigmoide.
ReLU	$\max(0, z)$	1 si $z > 0$, else 0	Evita el desvanecimiento del gradiente en los valores positivos.
Softmax	$\frac{e^{z_i}}{\sum e^{z_j}}$	Jacobiana compleja	Crucial para la clasificación multiclase.

Análisis de Sensibilidad mediante la Matriz Jacobiana

Mientras que el gradiente es un vector para funciones escalares, muchas operaciones en el aprendizaje profundo son funciones de valor vectorial, en las que una capa transforma un vector de entrada en otro. Para describir la sensibilidad completa de cada salida respecto de cada entrada, se utiliza la matriz jacobiana (J) (Sayin et al., 2025).

Estructura y Propiedades de la Jacobiana

Para una función $f : \mathbb{R}^n \to \mathbb{R}^m$, la Jacobiana es una matriz de dimensiones $m \times n$ donde el elemento en la fila i y columna j es la derivada parcial de la i-ésima salida con respecto a la j-ésima entrada:

$$J = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

En la retropropagación, la Jacobiana representa la transformación lineal local de la capa. El producto vector-Jacobiano (VJP) es el mecanismo matemático exacto mediante el cual el gradiente de la pérdida con respecto a las salidas de una capa se convierte en el gradiente con respecto a sus entradas. Frameworks como PyTorch y TensorFlow automatizan este proceso al rastrear las operaciones hacia adelante y aplicar los VJP correspondientes en orden inverso.

Estabilidad de la Jacobiana e Inicialización de Pesos

La estabilidad del entrenamiento depende críticamente de que las normas de las matrices jacobianas de cada capa no se desvíen excesivamente de la unidad. Si el producto de las Jacobianas a lo largo de muchas capas tiene valores singulares muy superiores a 1, los gradientes explotarán; si son muy inferiores a 1, los gradientes se desvanecerán, deteniendo el aprendizaje en las capas iniciales El análisis basado en el cálculo de estas matrices ha permitido desarrollar esquemas de inicialización, como la inicialización de Xavier o He, que garantizan que la varianza de las activaciones y los gradientes

se mantenga constante a través de la profundidad de la red.

La Matriz Hessiana y la Curvatura del Paisaje de Pérdida

Para comprender no sólo la dirección de descenso, sino también la eficiencia y estabilidad del entrenamiento, es necesario recurrir a las derivadas de segundo orden, capturadas en la matriz hessiana (H) La hessiana es una matriz cuadrada que contiene todas las derivadas parciales de segundo orden de la función de pérdida con respecto a los parámetros

Curvatura y el hiper parámetro de estabilidad

La hessiana describe la curvatura local de la superficie de pérdida. Sus autovalores (λ) indican cuán rápido varía el gradiente en distintas direcciones. Una dirección con un autovalor grande corresponde a una cañada estrecha con paredes empinadas (alta curvatura), mientras que un autovalor pequeño indica una región plana (baja curvatura) (Lee, 2023).

La estabilidad del descenso de gradiente está determinada por el autovalor máximo de la Hessiana, a menudo denominado nitidez (sharpness). Si el paso de actualización es mayor que la inversa de la nitidez, el algoritmo puede oscilar violentamente o incluso saltar fuera de la región de convergencia.

El fenómeno del borde de estabilidad (Edge of Stability)

Investigaciones recientes han revelado un comportamiento fascinante en el entrenamiento de redes neuronales profundas conocido como el Borde

de Estabilidad (EoS). Durante el entrenamiento con una tasa de aprendizaje constante η, se observa que la nitidez de la Hessiana aumenta progresivamente hasta alcanzar un umbral crítico de $2/\eta$. En este punto, el entrenamiento entra en una fase donde la pérdida deja de disminuir de forma monótona y comienza a oscilar, pero el algoritmo se auto-estabiliza manteniendo la nitidez cerca de este valor crítico.

Este aumento de nitidez se debe a la alineación de las matrices jacobianas de las capas. A medida que el modelo aprende, los vectores singulares de las Jacobianas de capas consecutivas se alinean, lo que significa que pequeños cambios en las entradas o en las capas iniciales provocan cambios masivos en la salida final de la red. Este fenómeno subraya cómo la dinámica del entrenamiento está dictada por las propiedades geométricas profundas del cálculo multivariable (véase la Tabla 15).

Tabla 15: Dinámica de entrenamiento y las propiedades geométricas

Característica de la hessiana	**Significado Geométrico**	**Implicación para el Entrenamiento**
Autovalores Grandes ($\lambda \gg 0$)	Curvatura alta (región afilada).	Requiere tasas de aprendizaje bajas para evitar la inestabilidad.
Autovalores Pequeños	Región plana (meseta).	El entrenamiento puede estancarse debido a

($\lambda \approx 0$)		gradientes casi nulos.
Autovalores Negativos ($\lambda < 0$)	Curvatura hacia abajo.	Indica la presencia de un punto de ensilladura o de un máximo local.
Nitidez (λ_{max})	Peor curvatura local.	Define el límite superior de la tasa de aprendizaje estable ($\eta < 2/\lambda_{max}$).

Cálculo de Variaciones y Operadores en Redes Convolucionales (CNN)

Las redes neuronales convolucionales (CNN) presentan un desafío matemático único: el uso de la operación de convolución para extraer características espaciales de imágenes y de cuadrículas de datos. Aunque computacionalmente eficiente, el análisis de sus gradientes requiere una comprensión matizada del cálculo aplicado a funciones definidas sobre dominios discretos.

La Dualidad Convolución-Correlación Cruzada

En el paso hacia adelante (forward pass), una CNN aplica un filtro o núcleo (kernel) K una imagen de entrada .I La operación, aunque a menudo

se denomina convolución en las bibliotecas de aprendizaje profundo, es técnicamente una correlación cruzada. El valor en la posición (i, j) del mapa de características resultante es:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n)$$

El cálculo multivariable revela una elegante dualidad durante el paso hacia atrás (backward pass). Para encontrar el gradiente de la pérdida con respecto a los pesos del filtro ($\frac{\partial L}{\partial K}$), el algoritmo realiza una operación de correlación cruzada entre la imagen de entrada y el gradiente proveniente de la capa siguiente. Por otro lado, para calcular el gradiente con respecto a la imagen de entrada ($\frac{\partial L}{\partial I}$) y así continuar la retropropagación, se debe realizar una convolución completa entre el gradiente de salida y el filtro original girado 180 grados. Esta rotación no es una decisión de diseño arbitraria, sino una consecuencia directa de las derivadas parciales de la operación de suma ponderada deslizante.

Invarianza Espacial y Compartición de Pesos

La compartición de pesos en las CNN, donde el mismo filtro se aplica a toda la imagen, significa que cada peso del filtro contribuye a múltiples píxeles de la salida Según la regla de la cadena multivariable, el gradiente total para un solo peso del filtro es la suma de las contribuciones de todos los píxeles de salida donde se aplicó ese peso Esta agregación de gradientes es lo que permite a las CNN aprender detectores de características que son efectivos independientemente de la posición de la característica en la imagen

Retropropagación a través del tiempo (BPTT) en redes recurrentes

Las redes neuronales recurrentes (RNN) extienden el cálculo multivariable al dominio temporal, lo que permite procesar secuencias de longitud variable. En una RNN, el estado oculto en el tiempo t t es una función del estado oculto anterior t−1 y de la entrada actual t.

Dependencias Temporales y la Cadena de Jacobianas

Debido a la estructura recursiva $h_t = f(h_{t-1}, x_t, W)$, un peso W influye en la pérdida final a través de su efecto en cada paso de tiempo individual y de cómo ese efecto se propaga a los estados futuros. La derivada total de la pérdida E con respecto a W se formula como una suma sobre el tiempo:

$$\frac{dE}{dW} = \sum_{t-1}^{T} \frac{\partial E}{\partial h_t} \frac{\partial h_t}{\partial W}$$

Donde $\frac{\partial E}{\partial h_t}$ representa el error acumulado desde el final de la secuencia hasta el tiempo t_4 Este término se calcula retropropagando a través de los pasos de tiempo, lo que implica multiplicar una cadena de matrices jacobianas de la función de transición de estado (Sandubete et al., 2023).

El problema de la explosión y desvanecimiento de gradientes

El análisis del cálculo en RNNs muestra que la derivada del estado oculto final respecto de un estado inicial da lugar a un producto de términos $\frac{\partial h_T}{\partial h_1} = \prod_{t=2}^{T} \frac{\partial h_t}{\partial h_{t-1}}$. Si los autovalores de estas matrices Jacobianas son consistentemente mayores que 1, el gradiente crecerá exponencialmente (explosión); si son menores que 1, decrecerá exponencialmente (desvanecimiento), impidiendo que la red aprenda dependencias a largo plazo El uso de funciones como la tangente hiperbólica (Tanh) ayuda a mitigar esto al acotar las activaciones, pero arquitecturas más complejas como LSTM o GRU se diseñaron específicamente basándose en estos principios de cálculo para crear vías rápidas donde el gradiente pueda fluir sin ser multiplicado repetidamente por valores pequeños.

Optimizadores Avanzados: Aproximaciones de Momentos de Orden Superior

Mientras que el descenso de gradiente simple utiliza solo la primera derivada, los optimizadores modernos como Adam, RMSprop y los métodos de segundo orden buscan aprovechar la información de la curvatura para acelerar la convergencia (Zaznov et al., 2025).

Adam y la Adaptación Paramétrica

El optimizador Adam (Adaptive Moment Estimation) es ampliamente utilizado por su robustez en entornos con funciones de pérdida complejas. Matemáticamente, Adam estima los dos primeros momentos de los gradientes: la media (primer momento) y la varianza no centrada (segundo momento).

1. **Estimación del primer momento:** que actúa como un promedio móvil del

gradiente para proporcionar inercia (momentum).

2. **Estimación del Segundo Momento:** que rastrea la magnitud de las oscilaciones de los gradientes.

Al actualizar los pesos, Adam escala el paso de aprendizaje de cada parámetro inversamente proporcional a la raíz cuadrada del segundo momento de ese parámetro. Esto significa que los parámetros con gradientes altamente volátiles reciben actualizaciones más cautelosas, mientras que aquellos en regiones planas reciben pasos más grandes, compensando de manera efectiva la falta de información de segundo orden directa sin el costo computacional de invertir la Hessiana.

Métodos de Segundo Orden y K-FAC

Para ir más allá de las aproximaciones de primer orden, se han desarrollado métodos como el de Newton o el de gradiente natural. El Gradiente Natural utiliza la métrica de información de Fisher en lugar de la Hessiana estándar para realizar actualizaciones invariantes a la reparametrización del modelo (Blanco y Cervera, 2003).

Una de las técnicas más prometedoras es la Curvatura Aproximada Factorizada de Kronecker (K-FAC). K-FAC aproxima la matriz de Fisher (o la Hessiana de Gauss-Newton generalizada, GGN) como un producto de Kronecker de matrices más pequeñas para cada capa, lo que permite su inversión eficiente. Este enfoque ha demostrado reducir significativamente el número de iteraciones necesarias para el entrenamiento, lo que se traduce en una reducción de 50-75% en el tiempo de computación en arquitecturas modernas como las Transformers.

Geometría y Topología de Paisajes de Pérdida en Alta Dimensión

La comprensión del éxito del aprendizaje profundo requiere mirar más allá de los puntos críticos locales y considerar la topología global de la superficie de pérdida. En espacios de millones de dimensiones, nuestra intuición basada en el mundo físico de tres dimensiones suele ser errónea

La proliferación de puntos de ensilladura

Históricamente, se temía que los mínimos locales con errores elevados fueran el principal obstáculo para el entrenamiento de redes profundas. Sin embargo, el análisis mediante cálculo multivariable y teoría de matrices aleatorias indica que, en altas dimensiones, es extremadamente improbable encontrar mínimos locales de alta energía. En su lugar, lo que abunda son los puntos de ensilladura (saddle points), donde la Hessiana tiene tanto autovalores positivos como negativos

En un punto de ensilladura, la superficie se curva hacia arriba en algunas direcciones y hacia abajo en otras. Mientras que el descenso de gradiente de primer orden puede estancarse temporalmente en las mesetas que rodean estos puntos, el algoritmo suele encontrar, con el tiempo, una dirección de curvatura negativa por la que escapar. Los métodos de segundo orden son teóricamente superiores para navegar por estas regiones, ya que pueden identificar y seguir activamente las direcciones de descenso más rápidas, incluso cuando el gradiente es casi nulo (Safari et al., 2020).

Paisajes Convexos vs. Caóticos

La arquitectura de la red tiene un impacto dramático en la suavidad de la superficie de pérdida. A medida que las redes se vuelven más profundas, el paisaje de pérdida tiende a transicionar de regiones casi convexas a superficies altamente caóticas y fractales. Esta transición suele coincidir con una caída de la capacidad de entrenamiento y un aumento del error de generalización. Técnicas como las conexiones residuales (skip connections) en las Reses han demostrado matemáticamente aplanar o suavizar este paisaje, permitiendo que el gradiente fluya de manera más predecible y facilitando la convergencia en redes de cientos de capas.

El Papel del Cálculo en la Generalización y la Robustez

El cálculo multivariable no solo explica cómo las redes neuronales minimizan el error en el conjunto de entrenamiento, sino que también ofrece pistas sobre por qué estos modelos funcionan tan bien con datos nuevos.

Mínimos Planos y el Principio de Estabilidad

Se ha observado empíricamente que los mínimos planos (aquellos en los que la hessiana tiene autovalores pequeños en la mayoría de las direcciones) tienden a generalizar mucho mejor que los mínimos afilados. Un plano mínimo significa que pequeñas perturbaciones en los parámetros (debidas al ruido de los datos o a la precisión numérica) no cambian drásticamente la salida del modelo, lo que indica una mayor robustez.

Algoritmos recientes, como la Minimización Consciente de la Nitidez (Sharpness-Aware Minimization, SAM), integran este concepto de cálculo

directamente en el ciclo de entrenamiento. En lugar de simplemente buscar un punto con pérdida mínima, SAM busca regiones donde la pérdida sea uniformemente baja en un vecindario de los parámetros, penalizando efectivamente las direcciones de alta curvatura en la matriz de Hessiana.

Regularización basada en jacobianas y hessianas

Para mejorar la robustez frente a ataques adversarios, se emplean técnicas de regularización que limitan explícitamente la norma de las matrices jacobianas y hessianas del modelo con respecto a sus entradas. Al reducir la magnitud de las derivadas parciales de primer y segundo orden, se asegura que el modelo sea menos sensible a pequeñas perturbaciones maliciosas en las imágenes o en los datos de entrada (Sariev y Germano, 2020). Este enfoque generaliza los esfuerzos previos de robustez, lo que permite construir clasificadores que no solo son precisos, sino también estables ante cambios en el entorno.

El análisis exhaustivo presentado demuestra que el cálculo multivariable no es solo un requisito académico para el estudio de la inteligencia artificial, sino también la herramienta fundamental que hace posible el aprendizaje en máquinas complejas. Desde la derivación de la retropropagación hasta el análisis de la estabilidad en el borde de nitidez, cada avance significativo en el campo ha estado anclado en principios de diferenciación y optimización.

La capacidad de calcular gradientes en grafos de computación masivos ha democratizado el aprendizaje profundo, permitiendo a investigadores y desarrolladores entrenar modelos con miles de millones de parámetros. Sin embargo, como muestra el estudio de los paisajes de pérdida y de la dinámica

de las Jacobianas, aún queda mucho por descubrir sobre la interacción entre la arquitectura del modelo y la topología de la optimización.

El futuro del campo probablemente resida en una integración aún más estrecha entre el cálculo geométrico y la ingeniería de sistemas, que busque optimizadores que no solo desciendan gradientes, sino que comprendan y naveguen activamente por la compleja curvatura del espacio de parámetros de alta dimensión. La transición hacia métodos de segundo orden más eficientes y la comprensión de la autoestabilización en el entrenamiento profundo son las fronteras en las que el cálculo multivariable seguirá definiendo los límites de lo que la inteligencia artificial puede alcanzar.

Capítulo 6

Arquitectura Geométrica y Transformaciones Trigonométricas en el Aprendizaje de Máquina

La intersección entre la geometría, la trigonometría y el aprendizaje automático constituye uno de los pilares más sofisticados de la ciencia de datos contemporánea. Lejos de ser meras herramientas de cálculo auxiliar, estas disciplinas proporcionan la fundamentación lógica y estructural necesaria para que los algoritmos puedan razonar sobre la forma y la periodicidad de los datos.

La geometría aporta una formación lógica que permite una comprensión profunda de las aplicaciones complejas, desde la aeronáutica hasta la ingeniería de sistemas masivos, mientras que la trigonometría, originada en las necesidades de la astronomía, se ha convertido en un instrumento imprescindible en áreas técnicas y científicas que requieren el manejo de ciclos y rotaciones. En el contexto del aprendizaje de máquina, este marco matemático permite la creación de fronteras de decisión robustas, la proyección de datos en variedades de baja dimensionalidad y la optimización de redes neuronales a través de paisajes de pérdida cuya topología define la capacidad de generalización del modelo

Fundamentos de Geometría Analítica y

Álgebra Lineal en el Espacio de Características

El aprendizaje de máquina opera fundamentalmente con representaciones vectoriales de la realidad. Cada observación se interpreta como un punto en un espacio de características de alta dimensionalidad, donde los entes geométricos fundamentales —punto, recta, plano y volumen— adquieren significados computacionales críticos. El punto, carente de dimensión, representa la posición de una instancia individual; la recta, con longitud pero sin ancho, define trayectorias de regresión o ejes de varianza; y el plano, con longitud y ancho, sirve como el separador canónico en problemas de clasificación (Liang et al., 2025).

La geometría plana y espacial permite extender estas nociones a dimensiones arbitrarias mediante el concepto de hiperplano. En un sistema de aprendizaje supervisado, un hiperplano se define mediante la ecuación general:

$$h(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p = x^T \beta + \beta_0 = 0$$

donde β es el vector normal al hiperplano y β_0 es el sesgo o bias. Esta estructura geométrica es la base de las Máquinas de Vectores de Soporte (SVM), que buscan identificar el hiperplano óptimo capaz de maximizar el margen entre las distintas clases. El clasificador de margen máximo se define como aquel hiperplano que maximiza la distancia M a las observaciones más cercanas de cada clase, denominadas vectores de soporte. Si el vector β está restringido a ser un vector unitario ($||\beta|| = 1$), entonces el producto del

hiperplano y la variable de respuesta $y_i \in \{-1, 1\}$ representa las distancias perpendiculares al hiperplano, lo que asegura que $y_i(x_i^T \beta + \beta_0) \geq M$.

Espacios Métricos y Geometría de la Distancia

La capacidad de un modelo para agrupar o clasificar datos depende de la métrica de distancia que se elija. La geometría euclidiana clásica, basada en el teorema de Pitágoras, define la distancia en línea recta entre dos puntos en un espacio multidimensional. No obstante, la distancia euclidiana es altamente sensible a la escala de los atributos y no considera la correlación intrínseca entre las variables (Ferreiros y Paz, 2023). Para superar estas limitaciones, se recurre a métricas más robustas, como la distancia de Mahalanobis, que normaliza la distancia en función de la varianza y la covarianza de los datos.

Geométricamente, el cálculo de la distancia de Mahalanobis implica una serie de transformaciones lineales: trasladar los datos al origen, rotarlos para eliminar las correlaciones (mediante la diagonalización de la matriz de covarianza) y estandarizar los datos para que todas las varianzas sean unitarias. Este proceso transforma una distribución elipsoidal en una distribución esférica, lo que permite una detección de valores atípicos mucho más precisa en espacios multivariados en los que las variables están fuertemente correlacionadas (véase la Tabla 16).

Tabla 16: Métrica de distancia y su aplicación en machine learning

Métrica de distancia	Formulación Matemática	Aplicación en Machine Learning	Sensibilidad a Escala/Outliers

Euclidiana (L_2)	$\sqrt{\sum(p_i - q_i$	Clustering K-means, KNN básico	Alta sensibilidad; requiere normalización
Manhattan (L_1)	$\sum	p_i - q_i	$
Mahalanobis	$\sqrt{(x - \mu)^T S^{-1}(x$	Detección de anomalías, clasificación robusta	Invariante a escala; considera correlación
Coseno	$\frac{A \cdot B}{		A
Chebyshev (L_∞)	$\max	p_i - q_i	$

Trigonometría en el Modelado de Fenómenos Periódicos y Secuenciales

La trigonometría proporciona las funciones necesarias para modelar la naturaleza cíclica de muchos conjuntos de datos reales. Las funciones seno y coseno, con su periodicidad de 2π radianes, son ideales para representar variables como el tiempo, las estaciones o las fases de un motor (Imbaquingo et al., 2024). En el aprendizaje de máquina, tratar estas variables de forma lineal (por ejemplo, asignar a las 23:00 y a las 00:00 valores extremos) introduce un sesgo geométrico, ya que el modelo percibe una distancia de 23 unidades cuando en realidad están separadas por una sola hora.

Codificación cíclica y transformaciones de Seno/Coseno

La solución técnica a este problema consiste en proyectar la variable sobre el círculo unitario. Para cualquier característica cíclica con periodo P, se generan dos nuevas características:

$$v_{\sin} = \sin\left(\frac{2\pi \cdot v}{P}\right)$$

$$v_{\cos} = \cos\left(\frac{2\pi \cdot v}{P}\right)$$

Esta transformación asegura que la distancia entre el final y el inicio del ciclo se conserve matemáticamente. Sin embargo, se debe tener precaución al utilizar modelos basados en árboles de decisión (Random Forest, XGBoost), ya que estos algoritmos realizan divisiones basadas en una sola variable a la vez, lo que puede dificultar la interpretación conjunta del par seno-coseno necesaria para entender la circularidad. Para redes neuronales y modelos lineales, esta codificación es el estándar de oro, permitiendo que el algoritmo aproxime funciones periódicas complejas con mayor facilidad.

Cálculo Trigonométrico y Análisis de Fourier

El cálculo de funciones trigonométricas proporciona la infraestructura matemática para la optimización basada en gradientes y el reconocimiento de patrones de frecuencia. Las derivadas de las funciones trigonométricas presentan una estructura cíclica que facilita el flujo de gradientes mediante la regla de la cadena en redes neuronales con activaciones sinusoidales.

El análisis de Fourier es una de las aplicaciones más profundas, ya que

permite descomponer señales en componentes de frecuencia mediante bases de senos y cosenos. En el aprendizaje de máquina, la Transformada Discreta de Fourier (DFT) y su implementación eficiente (FFT) permiten la extracción de características en el dominio de la frecuencia, fundamental para el procesamiento de audio y la visión artificial mediante espectrogramas, así como para métodos de kernel escalables como los Random Fourier Features.

Geometría de la Optimización y Paisajes de Pérdida

La optimización de un modelo de aprendizaje profundo puede visualizarse como la navegación por una superficie de alta dimensionalidad, denominada paisaje de pérdida (loss landscape). La topología de esta superficie determina no solo la velocidad de entrenamiento (convergencia), sino también la robustez del modelo resultante ante datos nuevos (generalización)

Curvatura y la matriz hessiana

Para analizar la geometría local de la función de pérdida respecto de los parámetros del modelo, se utiliza la matriz hessiana, que contiene las segundas derivadas parciales. La definitividad de esta matriz en un punto crítico permite clasificar la curvatura:

- **Definida positiva:** El punto es un mínimo local (curvatura positiva).
- **Definida negativa:** El punto es un máximo local.
- **Indefinida:** El punto es un punto de silla (saddle point), donde la superficie sube en algunas direcciones y desciende en otras.

En modelos complejos, la presencia de puntos de silla supone un desafío mayor que el de los mínimos locales. La optimización de Riemann aprovecha la estructura de variedad (manifold) del conjunto de restricciones para adaptar algoritmos de optimización no restringida y escapar de estos puntos mediante la información de curvatura. Esto es especialmente relevante en el ajuste fino de modelos, donde el conocimiento del Hessiano ayuda a adaptar las tasas de aprendizaje según la geometría local del espacio de parámetros

Impacto de las Conexiones de Salto y la Normalización

La arquitectura de la red tiene un efecto dramático en la suavidad del paisaje de pérdida. Las visualizaciones mediante direcciones aleatorias normalizadas por filtro han revelado que las redes profundas sin conexiones de salto (skip connections) presentan paisajes altamente caóticos e inconvexos, lo que dificulta enormemente su entrenamiento. Por el contrario, la introducción de conexiones residuales ($F(x) + x$) produce una transición hacia comportamientos casi convexos, lo que facilita que los algoritmos de descenso de gradiente encuentren mínimos de alta calidad (Wang et al., 2023).

Asimismo, el uso de capas de normalización (como Batch Normalization) ayuda a eliminar los efectos de la escala en los pesos de la red. Debido a la invariancia de escala de las funciones de activación como ReLU, multiplicar los pesos de una capa por un factor y dividir los de la siguiente por el mismo factor dejan la red inalterada, pero pueden cambiar drásticamente la magnitud del gradiente. La normalización estabiliza esta geometría, permitiendo que el optimizador ignore estas redundancias de escala y se concentre en la dirección del descenso (Sayin et al., 2025).

Geometría Diferencial y Manifold Learning

La hipótesis del manifold sugiere que los datos de alta dimensionalidad del mundo real (como imágenes o texto) residen en o cerca de variedades suaves de baja dimensionalidad. El aprendizaje de variedades (Manifold Learning) busca extraer estas geometrías intrínsecas para permitir un modelado más preciso y una reducción de la dimensionalidad que no comprometa las propiedades estructurales esenciales.

Distancia Geodésica vs. Euclidiana en Variedades No Lineales

Una distinción fundamental en esta área es la diferencia entre la distancia euclidiana y la distancia geodésica. Mientras que la distancia euclidiana es el segmento de recta que conecta dos puntos en el espacio ambiente, la distancia geodésica es la curva más corta que une ambos puntos y que permanece sobre la superficie de la variedad. En estructuras no lineales como el Swiss Roll, dos puntos pueden estar cerca en términos euclidianos pero muy alejados en términos geodésicos.

El algoritmo Isomap (Isometric Mapping) intenta preservar estas distancias geodésicas globales mediante la construcción de un grafo de vecinos cercanos y el cálculo de los caminos más cortos en él. Por otro lado, algoritmos como LLE (Locally Linear Embedding) se centran en preservar las relaciones lineales locales, asumiendo que cualquier variedad es aproximadamente plana en vecindarios lo suficientemente pequeños (Yao et al., 2017) (véase la Tabla 17).

Tabla 17: Algoritmos de Reducción de Dimensionalidad Geométrica

Algoritmo	Concepto Geométrico Clave	Fortalezas	Limitaciones
Isomap	Distancia Geodésica Global	Captura estructuras globales no lineales	Sensible al ruido y a los cortocircuitos en el grafo.
LLE	Linealidad Local	Preserva vecindarios locales; eficiente	Puede colapsar proyecciones si los pesos son inestables
t-SNE	Divergencia KL de vecindades	Excelente para visualización de clústeres	No preserva las distancias globales; es estocástico.
UMAP	Topología y Estructura Difusa	Preserva estructura local y global; rápido	Requiere un ajuste cuidadoso de los hiperparámetros.
MDS	Escalado Multidimensional	Mantiene distancias entre puntos	Principalmente lineal a menos que se use con kernels.

Geometría de la Información y Divergencias de Probabilidad

La geometría de la información trata el espacio de parámetros de un

modelo probabilístico como una variedad riemanniana. En este marco, las distribuciones de probabilidad son puntos en una superficie, y la diferencia entre ellas se mide no mediante distancias euclidianas, sino mediante divergencias que respetan la estructura de la información.

Divergencia de Kullback-Leibler y Entropía Relativa

La divergencia KL es la métrica más común para comparar una distribución real P con una aproximación. Geométricamente, la divergencia KL no es una distancia verdadera porque carece de simetría ($D_{KL}(P||Q) \neq D_{KL}(Q||P)$). Representa la cantidad de información (o sorpresa) adicional experimentada cuando se utiliza Q para modelar datos que en realidad siguen la distribución P. En el aprendizaje de máquina, minimizar la divergencia KL entre la verdad fundamental y las predicciones del modelo equivale a maximizar la log-verosimilitud, lo que guía el proceso de aprendizaje hacia soluciones óptimas.

Métrica de Fisher y Gradiente Natural

Cerca del máximo de verosimilitud, la curvatura de la función de pérdida está intrínsecamente relacionada con la información de Fisher. La Matriz de Información de Fisher (FIM) actúa como una métrica riemanniana en el espacio de parámetros, que indica cuán agudo o plano es el máximo de verosimilitud. Una información de Fisher alta implica que el modelo es muy sensible a pequeños cambios en los parámetros, mientras que una información de Fisher baja indica un máximo romo (blunt), en el que muchos valores de los parámetros producen resultados similares. El uso de esta métrica permite

implementar el gradiente natural, que actualiza parámetros invariantes a la reparametrización del modelo, lo que mejora significativamente la estabilidad en algoritmos de aprendizaje por refuerzo y en modelos generativos complejos.

Geometría de grafos y redes neuronales de grafos (GNN)

Los grafos pueden entenderse como variedades discretas cuya geometría local afecta la propagación de la información. Mientras que las redes neuronales tradicionales asumen una cuadrícula euclidiana (como los píxeles de una imagen), las GNN deben manejar topologías irregulares donde la noción de distancia y vecindad es puramente topológica

Curvatura de Ricci en grafos y el problema del cuello de botella

Un concepto emergente y poderoso es la aplicación de la curvatura de Ricci a los grafos. En geometría diferencial, la curvatura de Ricci determina el volumen del solapamiento entre dos bolas pequeñas; si es positiva, las bolas están más cerca entre sí que sus centros en términos de la distancia de transporte (Loisel y Romon, 2014). En un grafo, la curvatura de Ricci de Ollivier (ORC) mide el grado de solapamiento entre los vecindarios de dos nodos conectados por una arista:

- **Curvatura Positiva:** Indica comunidades densamente conectadas con muchos caminos alternativos (triángulos).
- **Curvatura Negativa:** Indica cuellos de botella o aristas tipo puente que

conectan clústeres diferentes, típicos en estructuras de árbol

El flujo de información en una GNN es análogo al flujo de calor en diversos sistemas. La tasa de difusión es más rápida en las aristas con curvatura positiva y más lenta en las con curvatura negativa. Modelos como RCGCN (Ricci Curvature-based GCN) integran esta información para adaptar la agregación de mensajes, mitigando problemas como el over-smoothing (donde las representaciones de los nodos se vuelven indistinguibles) y optimizando el aprendizaje en grafos con topologías complejas

Redes Neuronales Equivariantes a Grupos (G-CNN)

La geometría también determina cómo los modelos deben responder a las simetrías. Las CNN tradicionales son intrínsecamente equivariantes a la traslación, pero fallan ante rotaciones, reflexiones o escalados. Una función es equivariante si transformar la entrada y luego aplicar la función produce el mismo resultado que aplicar la función y luego transformar la salid (Awaluddin et al., 2023)

Las G-CNN (Group equivariant CNNs) utilizan la teoría de grupos para aprovechar simetrías adicionales y reducir la complejidad de las muestras. Mediante el uso de G-convoluciones, la red comparte pesos no solo a través de traslaciones, sino también a través de otros elementos de un grupo de simetría G (como rotaciones de 90 grados en el grupo $p4$). Esto aumenta la capacidad expresiva de la red sin incrementar el número de parámetros, lo que permite que el modelo aprenda representaciones robustas frente a transformaciones geométricas genéricas del mundo físico (véase la Tabla 18).

Tabla 18: Simetrías y propiedades en la red

Concepto de Simetría	Grupo Matemático	Propiedad en la red neuronal
Traslación	$\mathbb{Z}^2$	Equivarianza estándar de CNNs.
Rotación Discreta	C_4 (90º)	Levantamiento de convolución (lifting convolution).
Rotación Continua	$SO(2$	Redes neuronales Steerable.
Simetría Esférica	$SO(3$	CNNs esféricas para imágenes de 360°.
Permutación	S_n	Invariancia en Graph Neural Networks (GNNs).

Operadores Neuronales de Fourier y Continuidad Funcional

Una de las fronteras más recientes en la intersección entre la trigonometría y el aprendizaje profundo son los operadores neuronales de Fourier (FNO). A diferencia de las redes neuronales estándar que aprenden mapeos entre vectores de dimensión fija, los FNO aprenden mapeos entre espacios de funciones de dimensión infinita.

La arquitectura de un FNO se basa en la convolución espectral. Para una entrada dada, la capa realiza una transformada de Fourier, aplica una transformación lineal a los modos de baja frecuencia en el dominio de Fourier

y luego vuelve a realizar la transformada de Fourier inversa. Este enfoque tiene tres ventajas geométricas fundamentales:

1. **Invariancia a la Discretización:** El modelo puede entrenarse con datos a una resolución y evaluarse en una mucho mayor (zero-shot super-resolution) porque aprende la estructura continua de la función, no solo los valores en los puntos de la cuadrícula.
2. **Captura de Correlaciones Globales:** Dado que un cambio en el dominio de la frecuencia afecta a toda la señal en el dominio espacial, los FNO capturan dependencias de largo alcance de forma mucho más eficiente que las convoluciones locales.
3. **Eficiencia en EDP:** Son órdenes de magnitud más rápidos que los solucionadores numéricos tradicionales para ecuaciones diferenciales parciales (EDP) complejas, como las de Navier-Stokes en regímenes turbulentos, y mantienen una precisión comparable.

Recientemente, el desarrollo del Operador Neuronal de Vandermonde (VNO) ha extendido estas capacidades a distribuciones de puntos no equiespaciados, mediante matrices de Vandermonde estructuradas para computar transformadas de Fourier en dominios arbitrarios, lo que abre la puerta al aprendizaje de operadores en geometrías irregulares y en manifolds complejos.

La integración de la geometría y la trigonometría en el aprendizaje de máquina ha dejado de ser un mero complemento estético y se ha convertido en una necesidad funcional para la próxima generación de sistemas de inteligencia artificial. La transición del procesamiento de datos puramente euclidianos al aprovechamiento de variedades riemannianas, simetrías de

grupo y curvaturas de grafos permite abordar problemas que antes se consideraban intratables debido a su alta dimensionalidad o complejidad estructural.

El futuro del campo parece dirigirse hacia la Geometría Profunda, donde las arquitecturas de red no solo se inspiran en la geometría, sino que están estrictamente restringidas por ella para garantizar la consistencia física y la generalización matemática. La capacidad de los modelos para operar sobre complejos simpliciales —que capturan no solo relaciones entre pares de nodos, sino también relaciones de orden superior entre celdas, volúmenes y ángulos diedros— promete una nueva era en la interpretabilidad de los modelos. Al entender el aprendizaje de máquina como un proceso de transformación y partición geométricas del espacio, los investigadores pueden diseñar sistemas que no solo ajustan curvas a los datos, sino que también capturan la esencia topológica de la realidad que intentan modelar.

Conclusión

Un análisis detallado de los fundamentos y la formulación del problema muestra que la inteligencia artificial va más allá de ser solo una rama de la informática; es una convergencia interdisciplinaria que requiere un conocimiento profundo de las matemáticas. Considerar el Machine Learning como una caja negra no solo limita la innovación individual, sino que también genera vulnerabilidades en el sistema, como sesgos, ineficiencias y una mayor dependencia tecnológica.

Este libro de investigación surge como una respuesta fundamental a la crisis del entendimiento teórico. Al combinar álgebra lineal, cálculo, probabilidad y optimización en un enfoque narrativo orientado a resolver problemas reales, se pretende reducir la brecha de habilidades que actualmente separa a los ingenieros prácticos de los investigadores más avanzados. Para regiones como América Latina, esta formación no es un lujo académico, sino una necesidad crucial para superar las trampas del desarrollo y participar de manera equitativa en la soberanía digital mundial.

Uno de los mayores obstáculos para la soberanía digital en la región es la acelerada fuga de especialistas. La brecha de talento respecto al promedio mundial se ha ensanchado desde 2022, ya que los profesionales más capacitados suelen ser reclutados por empresas tecnológicas en Estados Unidos o en Europa. Aquellos que permanecen en la región a menudo se enfrentan a una formación alfabetizada pero poco especializada; es decir, a personas que saben usar la IA pero no saben cómo construirla ni adaptarla a los problemas locales, como la agricultura de precisión en climas tropicales o

la gestión de recursos hídricos en los Andes.

Además, persiste una brecha digital entre las áreas urbanas y rurales. Sin electricidad confiable, acceso a la banda ancha y dispositivos adecuados, millones de personas quedan relegadas en la revolución de la IA. Esta desigualdad no solo es injusta, sino que también limita la región, privándola de una variedad de datos única: los datos de ciudades y zonas rurales latinoamericanas son fundamentales para entrenar modelos que comprendan y afronten las trampas del desarrollo señaladas por la CEPAL, como la baja movilidad social y la debilidad institucional.

Frente a estas barreras, el software de código abierto (Open Source) surge como una oportunidad estratégica para América Latina. Alrededor del 38% de las organizaciones latinoamericanas ya utilizan IA de código abierto, lo que permite a los desarrolladores adquirir habilidades de vanguardia de manera asequible y auditable. Por ende, la colaboración público-privada es esencial para convertir este entusiasmo cultural en una industria de innovación sostenible y centrada en las personas.

Bibliografía

Akter, S., & Haider, MR (2025). mTanh: Una función de activación tolerante a gradientes evanescentes impresa con inyección de tinta de bajo costo. *Journal of Low Power Electronics and Applications* , *15* (2), 27. https://doi.org/10.3390/jlpea15020027

Awaluddin, B.A., Chao, C.T., & Chiou, J.S. (2023). Investigación de la transformación geométrica efectiva para el aumento de imágenes con el fin de mejorar los gestos manuales estáticos mediante una red neuronal convolucional preentrenada. *Mathematics* , *11* (23), 4783. https://doi.org/10.3390/math11234783

Barry, A., Li, W., Becerra, J. A., & Gilabert, P. L. (2021). Comparison of Feature Selection Techniques for Power Amplifier Behavioral Modeling and Digital Predistortion Linearization. *Sensors (Basel, Switzerland)*, *21*(17), 5772. https://doi.org/10.3390/s21175772

Blanco, E., y Cervera, M. (2002). *Mecánica de Estructuras.* Barcelona: Edicions UPC

Blume, S., Benedens, T., & Schramm, D. (2021). Técnicas de optimización de hiperparámetros para el diseño de sensores de software basados en redes neuronales artificiales. *Sensors* , *21* (24), 8435. https://doi.org/10.3390/s21248435

Brito Dumancela, C. A., Cargua Suarez, S. R., Vera Vera, M. V., & Gualsaquí Valencia, E. A. (2025). Álgebra lineal avanzada y su rol en el aprendizaje de algoritmos de machine learning. *Reincisol.*, *4*(7), 3440–3466. https://doi.org/10.59282/reincisol.V4(7)3440-3466

Cavani, M. (2025). Sobre un Recorrido de Estudio e Investigación del Álgebra

Lineal Necesaria para Reducir la Dimensionalidad con Machine Learning. *Revista de Investigación y Evaluación Educativa*, *12*(1), 27-47. https://doi.org/10.47554/revie.vol12.num1.2025.pp27-47

De la Cruz Serrano, F. (2025). Más allá de la fila por columna: intervención con GeoGebra y ETM en el aprendizaje del producto matricial en secundaria. *Quintaesencia*, *17*(1), 47–53. https://doi.org/10.54943/rq.v17i1.794

Ferreirós, J., y Paz, M. (2023). *La génesis de la geometría.* Madrid: Plaza y Valdés Editores. https://www.scielo.org.mx/pdf/rhfi/v57n169/0011-1503-rhfi-57-169-165.pdf

Gallego, E., Gómez-Ramírez, D.A.J., & Porras, J. (2025). Análisis topológico de datos y su aplicación en mercados financieros. *Revista Integración*, *43*(2), 15-26. https://doi.org/10.18273/revint.v43n2-2025002

Gao, L., Luo, Z., & Wang, L. (2025). Técnicas de aceleración de redes neuronales convolucionales basadas en plataformas FPGA: principios, métodos y desafíos. *Information* , *16* (10), 914. https://doi.org/10.3390/info16100914

García, Andrea, Restrepo, Ángela, & Velásquez, Juan D.. (2013). Búsqueda aleatoria repetitiva basada en caos. *Revista Ingenierías Universidad de Medellín*, *12*(22), 137-146. http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1692-33242013000100013&lng=en&tlng=es.

González, J.E. (2025). Pensamiento crítico y futuro sostenible: Una mirada desde Latinoamérica y el Caribe. Cartagena de Indias: Universu Editores. http://repositorio.unisinucartagena.edu.co:8080/jspui/bitstream/123456789/1829/1/Libro.Pensamiento_cr%C3%ADtico_y_futuro_sostenible._11agosto%5B1%5D.pdf

Guzmán Roldán, C. M., Estrada Huancas, M. M., Burga Barboza, R. E., & Villegas Santamaría, L. M. (2025). El Impacto de la Inteligencia Artificial en el Álgebra Lineal. Una revisión bibliométrica. *Revista Reflexiones De La Sociedad Y Economía*, *2*(2), 01–24. https://doi.org/10.62776/rse.v2i2.48

Hou, C. K. J., & Behdinan, K. (2022). Dimensionality Reduction in Surrogate Modeling: A Review of Combined Methods. *Data science and engineering*, *7*(4), 402–427. https://doi.org/10.1007/s41019-022-00193-5

Huang, J., Xu, Y., Wang, Q., Wang, QC, Liang, X., Wang, F., Zhang, Z., Wei, W., Zhang, B., Huang, L., Chang, J., Ma, L., Ma, T., Liang, Y., Zhang, J., Guo, J., Jiang, X., Fan, X., An, Z., Li, T., … Fei, A. (2025). Modelos fundamentales y toma de decisiones inteligente: progreso, desafíos y perspectivas. *Innovation (Cambridge (Mass.))*, *6* (6), 100948. https://doi.org/10.1016/j.xinn.2025.100948

Imbaquingo Guerrero, J. A., Bastidas González , K. A., Gutiérrez Bastidas , J. O., & Alvarado Rosado , S. M. (2024). Aplicación de trigonometría en la resolución de problemas de la vida cotidiana para estudiantes de bachillerato. *Reincisol.*, *3*(5), 1593–1607. https://doi.org/10.59282/reincisol.V3(5)1593-1607

Kim, J., & Lim, J. (2021). A Deep Neural Network-Based Method for Prediction of Dementia Using Big Data. *International journal of environmental research and public health*, *18*(10), 5386. https://doi.org/10.3390/ijerph18105386

Kim, J.E. (2025). Un enfoque de números hiperduales para el cálculo de derivadas de orden superior. *Axioms* , *14* (8), 641. https://doi.org/10.3390/axioms14080641

Kitao, A. (2022). Análisis de componentes principales y métodos relacionados para investigar la dinámica de macromoléculas biológicas. *J* , *5* (2), 298-317.

https://doi.org/10.3390/j5020021

Lee, M. (2023). The Geometry of Feature Space in Deep Learning Models: A Holistic Perspective and Comprehensive Review. *Mathematics*, *11*(10), 2375. https://doi.org/10.3390/math11102375

Liang, B., Wang, Y., & Tong, C. (2025). Razonamiento de IA en la era del aprendizaje profundo: De la IA simbólica a la IA neurosimbólica. *Mathematics* , *13* (11), 1707. https://doi.org/10.3390/math13111707

Loisel, B., & Romon, P. (2014). Ricci Curvature on Polyhedral Surfaces via Optimal Transportation. *Axioms*, *3*(1), 119-139. https://doi.org/10.3390/axioms3010119

Machine mediante Algoritmos Evolutivos [Trabajo fin de Master]. Universidad Nacional de Educación a Distancia (UNED)

Mishra, A., Jatt, V., Sefene, E. M., Salunkhe, S., Cep, R., & Abouel Nasr, E. (2025). Supervised Machine Learning and Physics Machine Learning approach for prediction of peak temperature distribution in Additive Friction Stir Deposition of Aluminium Alloy. *PloS one*, *20*(4), e0309751. https://doi.org/10.1371/journal.pone.0309751

Moyano-Arias, R. J., Salazar-Alvarez, E. G., & Toalombo-Vargas, V. M. (2024). El rol del Álgebra lineal en el desarrollo de algoritmos de machine learning. *MQRInvestigar*, *8*(4), 3693–3718. https://doi.org/10.56048/MQR20225.8.4.2024.3693-3718

Murad, A., Kraemer, FA, Bach, K., & Taylor, G. (2021). Aprendizaje profundo probabilístico para cuantificar la incertidumbre en la predicción de la calidad del aire. *Sensors* , *21* (23), 8009. https://doi.org/10.3390/s21238009

Oviedo Rodríguez, K., & Jiménez Oviedo., B. (2022). Basic examples of linear

algebra with python: Ejemplos básicos de álgebra lineal con python. *Revista Digital: Matemática, Educación E Internet*, *21*(1). https://doi.org/10.18845/rdmei.v21i1.5340

Pinto, J.M. (2023). *Optimización de parámetros en Extreme Learning*

Raschka, S., Patterson, J., & Nolet, C. (2020). Aprendizaje automático en Python: principales desarrollos y tendencias tecnológicas en ciencia de datos, aprendizaje automático e inteligencia artificial. *Information* , *11* (4), 193. https://doi.org/10.3390/info11040193

Ríos-Vásquez, G., & de la Fuente-Mella, H. (2023). Análisis matemático y modelización de los factores que determinan la calidad de vida en los municipios de Chile. *Matemáticas* , *11* (5), 1218. https://doi.org/10.3390/math11051218

Safari, M., Alves de Sousa, R., & Joudaki, J. (2020). Fabricación de superficies en forma de silla de montar mediante un proceso de conformado láser: una investigación experimental y estadística. *Metals* , *10* (7), 883. https://doi.org/10.3390/met10070883

Sandubete, JE, Beleña, L., & García-Villalobos, JC (2023). Prueba de la hipótesis del mercado eficiente y la paradoja del modelo-datos del caos en las principales divisas del mercado de divisas (FOREX). *Mathematics* , *11* (2), 286. https://doi.org/10.3390/math11020286

Sariev, E., & Germano, G. (2020). Redes neuronales artificiales bayesianas regularizadas para la estimación de la probabilidad de incumplimiento. *Quantitative Finance* , *20* (2), 311–328. https://doi.org/10.1080/14697688.2019.1633014

Sayın, KA, Gürsoy, NK, Yolcu, T., & Gürsoy, A. (2025). Sobre la sinergia de los optimizadores y las funciones de activación: un estudio comparativo de

CNN. *Mathematics* , *13* (13), 2088. https://doi.org/10.3390/math13132088

Stanimirović, P. S., Ćirić, M., Mourtas, S. D., Milovanović, G. V., & Petrović, M. J. (2024). Simultaneous Method for Solving Certain Systems of Matrix Equations with Two Unknowns. *Axioms*, *13*(12), 838. https://doi.org/10.3390/axioms13120838

Sun, G., Li, Z., Jiao, Y., & Wang, Q. (2025). Aplicación de la estadística bayesiana en el análisis y la predicción de los cambios dimensionales inducidos por la carburación en barras de torsión. *Metals* , *15* (5), 546. https://doi.org/10.3390/met15050546

Tian, Y., Zhang, Y., & Zhang, H. (2023). Avances recientes en el descenso de gradiente estocástico en el aprendizaje profundo. *Mathematics* , *11* (3), 682. https://doi.org/10.3390/math11030682

Vinegoni, C., Fumene Feruglio, P., Courties, G. et al. (2020). Representaciones de imágenes tensoriales de microscopía de fluorescencia para el análisis de conjuntos de datos a gran escala. *Sci Rep.*,10, 5632. https://doi.org/10.1038/s41598-020-62233-2

Wang, H., Yao, L., Wang, H., Liu, Y., Li, Z., Wang, D., Hu, R., & Tao, L. (2023). Supervised Manifold Learning Based on Multi-Feature Information Discriminative Fusion within an Adaptive Nearest Neighbor Strategy Applied to Rolling Bearing Fault Diagnosis. *Sensors*, *23*(24), 9820. https://doi.org/10.3390/s23249820

Yao, B., Su, J., Wu, L., & Guan, Y. (2017). Modified Local Linear Embedding Algorithm for Rolling Element Bearing Fault Diagnosis. *Applied Sciences*, *7*(11), 1178. https://doi.org/10.3390/app7111178

Yolles, M. (2022). Conciencia, sapiencia y sensibilidad: una perspectiva metacibernética. *Systems* , *10* (6), 254.

https://doi.org/10.3390/systems10060254

Zaznov, I., Badii, A., Kunkel, J. et al. (2025). AdamZ: un método de optimización mejorado para el entrenamiento de redes neuronales. *Neural Comput & Applic.* 37, 26887–26914. https://doi.org/10.1007/s00521-025-11649-w

Zhu, Y., Wang, M., Yin, X., Zhang, J., Meijering, E., & Hu, J. (2022). Deep Learning in Diverse Intelligent Sensor Based Systems. *Sensors (Basel, Switzerland)*, *23*(1), 62. https://doi.org/10.3390/s23010062

De esta edición de *“Matemáticas para el aprendizaje de máquina”*, se terminó de editar en la ciudad de Colonia del Sacramento en la República Oriental del Uruguay el 28 de febrero de 2026

EST. 2021 EMC
EDITORIAL MAR CARIBE

Matemáticas para el aprendizaje de máquina

Ángel Amado Romero Cahuana
Rosa Luz Medina Aguilar
Edinson Raúl Montoro Alegre
Domingo Guzmán Chumpitaz Ramos
Juan Honorato Luna Valdez
Olmedo Pizango Isuiza

ISBN: 978-9915-698-74-8
9 789915 698748