Inteligencia Artificial (AI) Usando Maquinas de Aprendizaje: Aplicaciones en Salud y Geociencias.

Pablo Guillen-Rondon, Ph.D.

Adjunct Faculty – University of Houston Downtown, Houston, TX, USA
R&D – University of Tulsa, TUFFP, OK, USA
Sr. Data Scientist, KlinicSync, Merida, Mexico
Professor – MIU, Research Group: Knowledge Society, UNIR, Spain

- B.Sc. Mathematics · M.Sc. Applied Mathematics
- M.Sc. Automation · Ph.D. Biomedical Engineering
- Postdoc — Computational Science, UT El Paso
- 25+ years R&D: Oil & Gas · Biomedical Sciences
- 135+ publications · Keynote Speaker

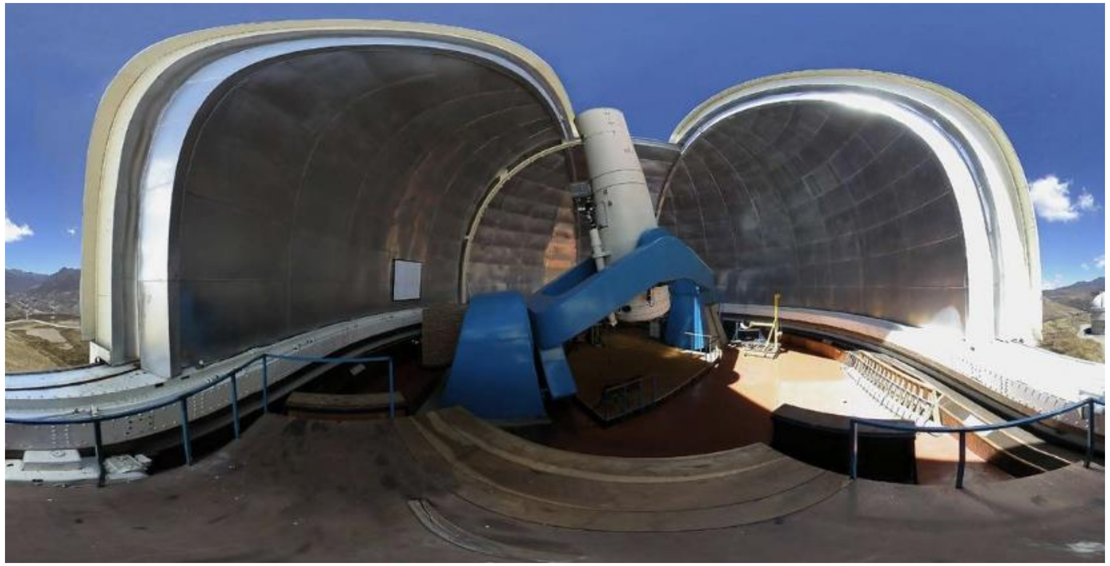
CUDI, Mexico, 2026

Astrometry

Bachelor THESIS — MATHEMATICS | UNIVERSITY OF THE ANDES, VENEZUELA | 1992

Characteristics of the Objective Prism of the C.I.D.A.

Field Distortion, Dispersion Curve & Astrometric Position Measurement with Objective-Prism Spectroscopy



CIDA — Centro de Investigaciones de Astronomia 'Francisco J. Duarte' Observatorio Nacional de Llano del Hato

INTRODUCTION

CIDA Schmidt Telescope

Focal length:	**3.010 m**
Mirror Ø:	**1.5 m**
Corrector Ø:	**1.0 m**
Plate size:	**30 × 30 cm**
Prism angle:	**3°.20′**
Prism material:	**UBK 7 (UV)**
Scale:	**16 µm/"**

Objective Prism Spectroscopy at CIDA

- Objective prisms dispersed stellar light across photographic plates, enabling simultaneous spectral classification of thousands of stars.
- The CIDA 1m Schmidt Camera used a UBK 7 glass prism (UV-transparent) placed in front of the telescope aperture.
- The resulting spectra allowed identification of emission-line stars, radial velocity measurements, and spectral type determination.
- A fundamental challenge: the prism introduces positional distortions that must be characterised and corrected for accurate astrometry.
- This thesis developed a novel mathematical theory to precisely model and correct that field distortion.

PHYSICS-INFORMED MATHEMATICAL MODEL

Spherical trigonometry + Snell's Law → exact prismatic field distortion

Snell's Law (Refraction)

Light deviation at each prism surface: $\sin \Omega_2 = n \cdot \sin W_2$.
Links incidence angle, refractive index $n(\lambda)$, and exit angle across both surfaces.

Spherical Trigonometry

Star position S_1 mapped on the celestial sphere relative to prism normals P_1 & P_2. Great-circle arcs used to propagate rays through the prism exactly.

Concentric Projection

Schmidt plates curve during exposure; arc-length $r = F \cdot \rho$ is preserved when plates flatten. This projection links plate coordinates (X,Y) to equatorial (α, δ).

Key Result: $\Delta\xi = c_{10}X + c_{01}Y + K \cdot r^2$ where $K = 84.92\ \mu m$ (distortion coefficient for $\lambda = 3970.1$ Å)

ORDINARY LEAST SQUARES (OLS) ESTIMATION

Polynomial distortion coefficients determined by minimising the sum of squared residuals

OLS Workflow

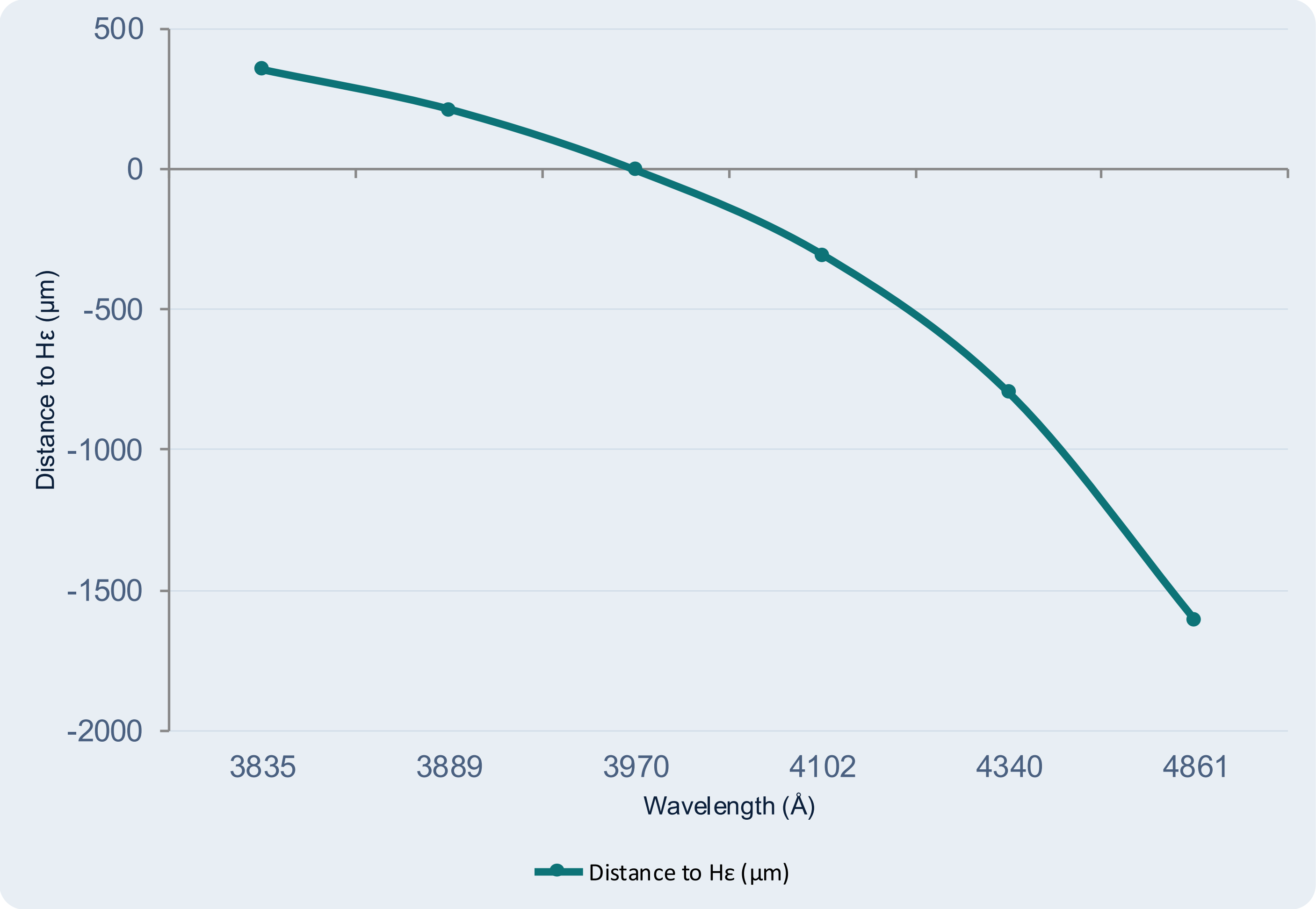
1. Simulate field of 1,640 equidistant stars using the physics-informed prism model (Prism1 program).
2. Compute distorted (X′, Y′) and undistorted (X, Y) Cartesian coordinates for the full plate field.
3. Express distortion as polynomial: $\Delta\xi = a_{00} + a_{10}X + a_{01}Y + a_{11}XY + a_{20}X^2 + a_{02}Y^2$.
4. Apply OLS (normal equations) to find coefficients a_ij minimising $\Sigma(\text{residuals})^2$.
5. Reduce to simplified form $\Delta\xi = c_{10}X + c_{01}Y + Kr^2$ using physical symmetry analysis.
6. Validate: max residual $R\Delta\xi$ = 0.15 µm — well below the 0.3 µm precision limit.

Theoretical Distortion Coefficients (CIDA)

Term	Δξ coefficient	Max effect (µm)
1 (const)	-0.1591×10^{-1}	0.02
X	-0.3491×10^{-3}	34.91
Y	$+0.4921 \times 10^{-3}$	49.21
XY	-0.6798×10^{-14}	0.00
X^2	$+0.8486 \times 10^{-8}$	84.86
Y^2	$+0.8502 \times 10^{-8}$	85.02
K (r^2)	**0.8492×10^{-8}**	**84.92 µm**

DISPERSION CURVE OF THE CIDA OBJECTIVE PRISM

Hartmann formula: wavelength vs. position — calibrated on 6 Hydrogen absorption lines



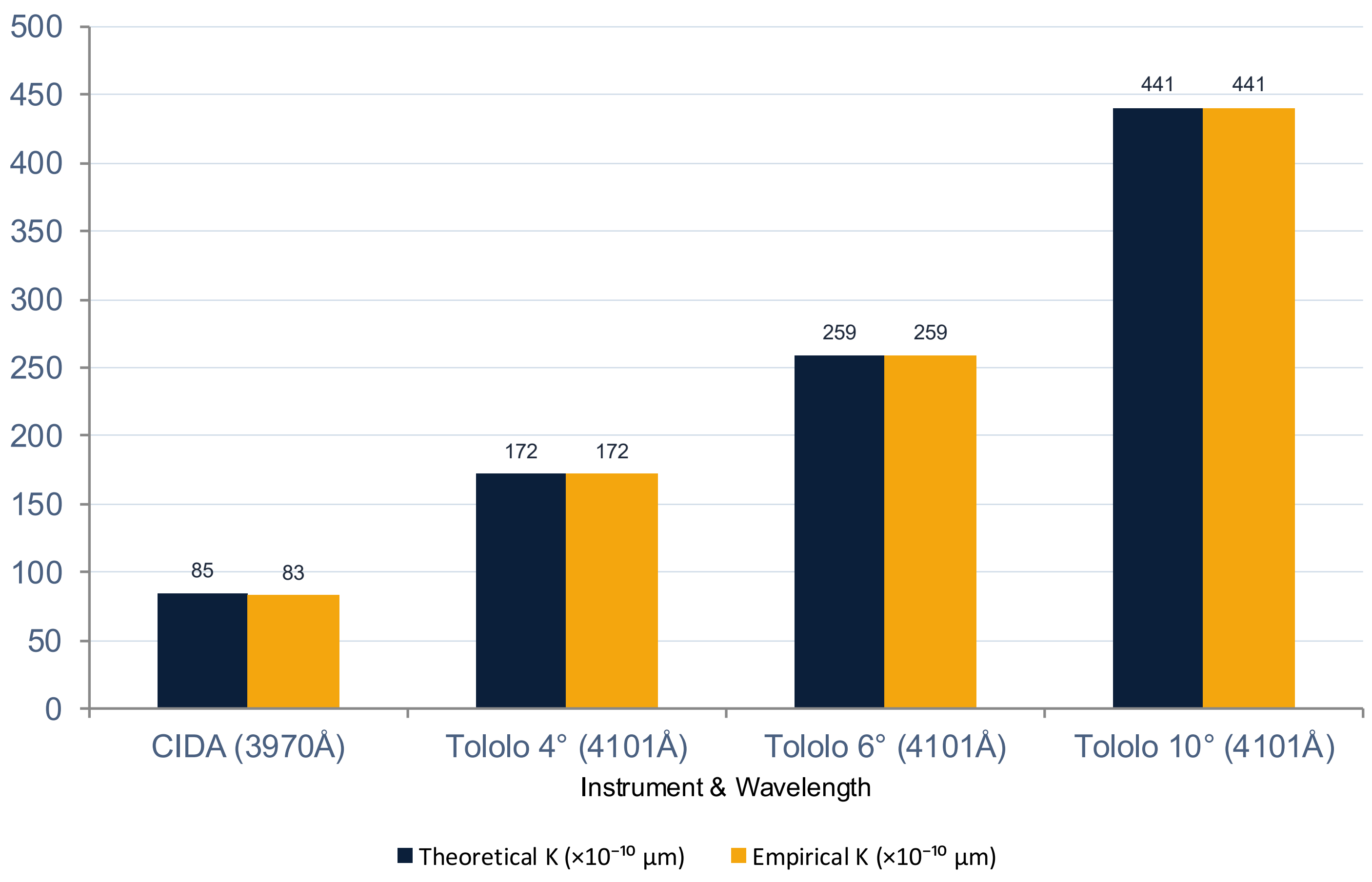
Key Calibration Results

Reference line	**Hε at 3970.1 Å**
λ range covered	**3835 – 4861 Å**
Plate material	**Kodak IIaO**
Hartmann constants	**λ_0=1616.6; X_0=−5859.5**
Dispersion at Hε	**401.4 Å/mm**
Velocity factor	**K = 82.78 × 10^{-10} µm (empirical)**
Theoretical K	**84.92 × 10^{-10} µm**
Agreement	**< 3% — theory confirmed**

Dispersion formula: $\lambda - \lambda_0 = a / (X - X_0)$ → $d\lambda/dX = -a/(X-X_0)^2$

EMPIRICAL VALIDATION

Theory vs. experiment: comparing prism plate and direct plate measurements



CIDA Agreement

Theoretical K = 85×10^{-10} µm vs empirical K = 83×10^{-10} µm. Difference < 3% — theory confirmed.

Cerro Tololo Prisms

For 4°, 6° & 10° prisms, theoretical distortion coefficients match Stock's empirical values exactly.

Residuals below limit

4° & 6° prisms: all residuals ≤ 0.3 µm. 10° prism: residuals up to 2 µm at field edge — a 3rd-degree polynomial would further improve this.

ASTROMETRIC REDUCTION METHODS

Converting measured plate coordinates (X, Y) → equatorial coordinates (α, δ)

Planar Method

(α, δ) → Projection → (ξ, η) → Polynomials → (X, Y)

- Transforms equatorial catalogue coordinates to theoretical plate coords via tangential or concentric projection.
- Relates theoretical (ξ, η) to measured (X, Y) through low-order polynomials capturing scale, rotation, and translation.
- Suitable for fields away from the Pole; computationally straightforward.
- Instrumental errors (origin shift, rotation w, scale factor P, axis obliquity ε) modelled explicitly.

Spherical Method

3D unit-sphere approach with orthogonal rotation matrix A

- Treats the sky as a unit sphere with 3D Cartesian coordinates (ξ, η, ζ) and plate coordinates (U, V, W).
- Connects both frames by a 3×3 orthogonal rotation matrix A (angles A, B, C), determined via OLS from reference stars.
- Higher-order distortion terms C_ij, D_ij added as corrections: $U = U' + \Sigma C_{ij} U'^{i} V'^{j}$.
- Non-orthogonality of A reveals residual distortions and scale errors from the prism.

RESULTS & CONCLUSIONS

A complete, physics-grounded framework for objective prism astrometry

①

Novel Prism Theory

New spherical-trigonometry formulation of objective prism optics, more precise than all prior treatments (Schwarzschild 1913, Fehrenbach 1947, Stock & Upgren 1968).

②

Physics-Informed Distortion Model

Distortion coefficients derived analytically from prism angle, mount parameters, and refractive index — no ad hoc fitting required.

③

Simplified OLS Polynomial

Full 2nd-degree polynomial reduces to $\Delta\xi = c_{10}X + c_{01}Y + Kr^2$ by physical symmetry, needing only 3 constants from plate data.

④

Dispersion Curve

Hartmann formula calibrated on 6 H-lines; dispersion = 401.4 Å/mm at Hε. Velocity factor enables radial velocity measurements from ΔX shifts.

⑤

Empirical Confirmation

Theoretical $K = 85\times10^{-10}$ µm agrees with empirical $K = 83\times10^{-10}$ µm within 3%. CTIO prisms also matched Stock's measurements exactly.

⑥

Enabling New Discoveries

The calibrated instrument enabled the 1993 CIDA Schmidt Survey (Briceño et al.), discovering 11 new T Tauri stars in Taurus-Auriga molecular clouds.

Pablo Guillén Rondón | University of the Andes, Venezuela | 1992 | Advisor: Dr. J. Stock

Methodological Contribution

- First complete physics-informed model for objective-prism field distortion at CIDA.
- Rigorous derivation using Snell's Law + spherical trig replaces empirical guesswork.
- OLS-reduced 3-parameter polynomial: practical for observers worldwide.

Instrumental Calibration

- Validated against Cerro Tololo Inter-American Observatory prisms — universal applicability demonstrated.
- Dispersion curve and velocity factor enable radial velocity science from prism plates.
- Precision 0.3 µm threshold maintained across the full 30×30 cm field.

Astronomical Discoveries

- Directly enabled the CIDA Schmidt H-alpha Survey (Briceño et al. 1993, PASP 105:686).
- Survey uncovered 11+ new T Tauri (pre-main sequence) stars in Taurus-Auriga.
- Paper received 39 citations — novel technique became a foundation for young-star surveys.

"Knowing the distortion caused by the prism in the position of images, the measured Cartesian coordinates are corrected to determine stellar positions with astrometric precision and radial velocity."

Biomedical application using symbolic dynamics

Depth of Anesthesia Monitoring

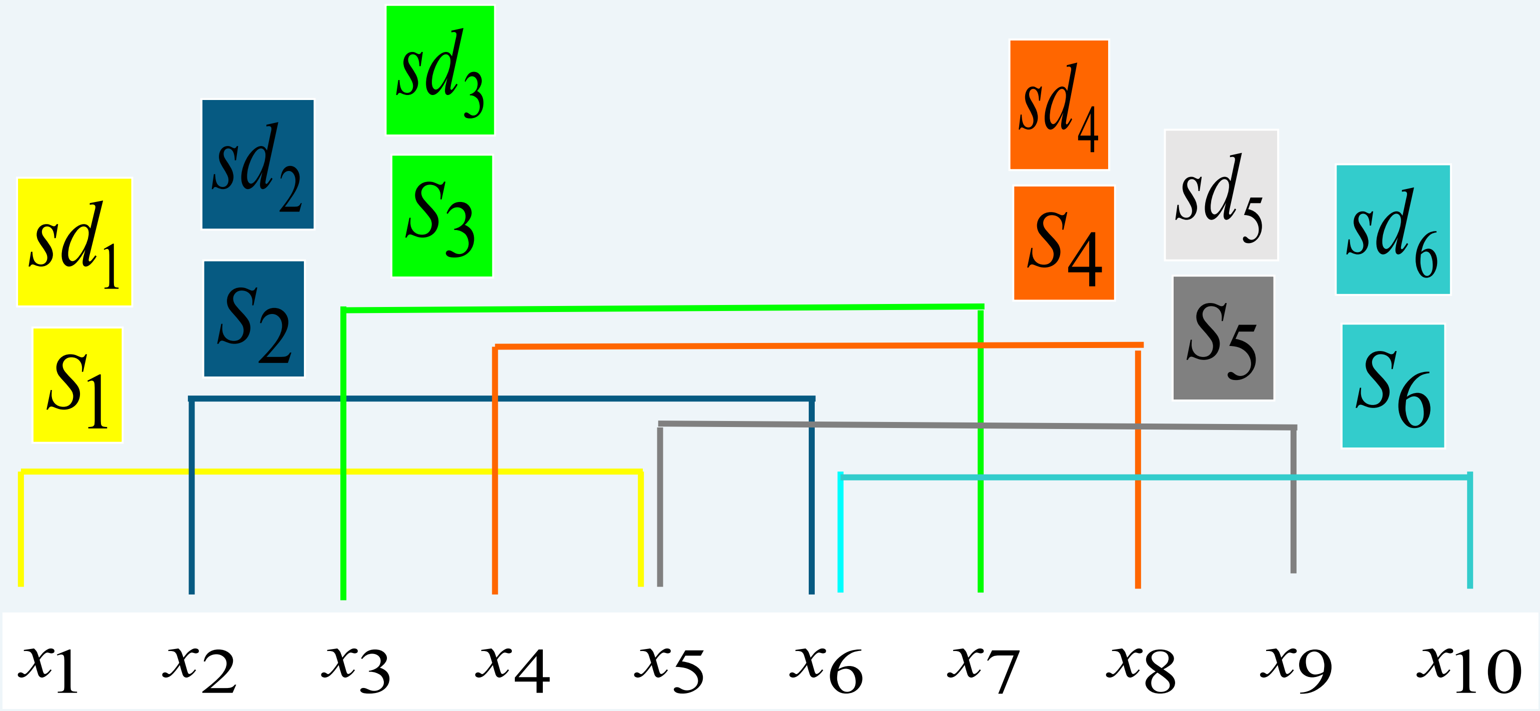
Abstract.

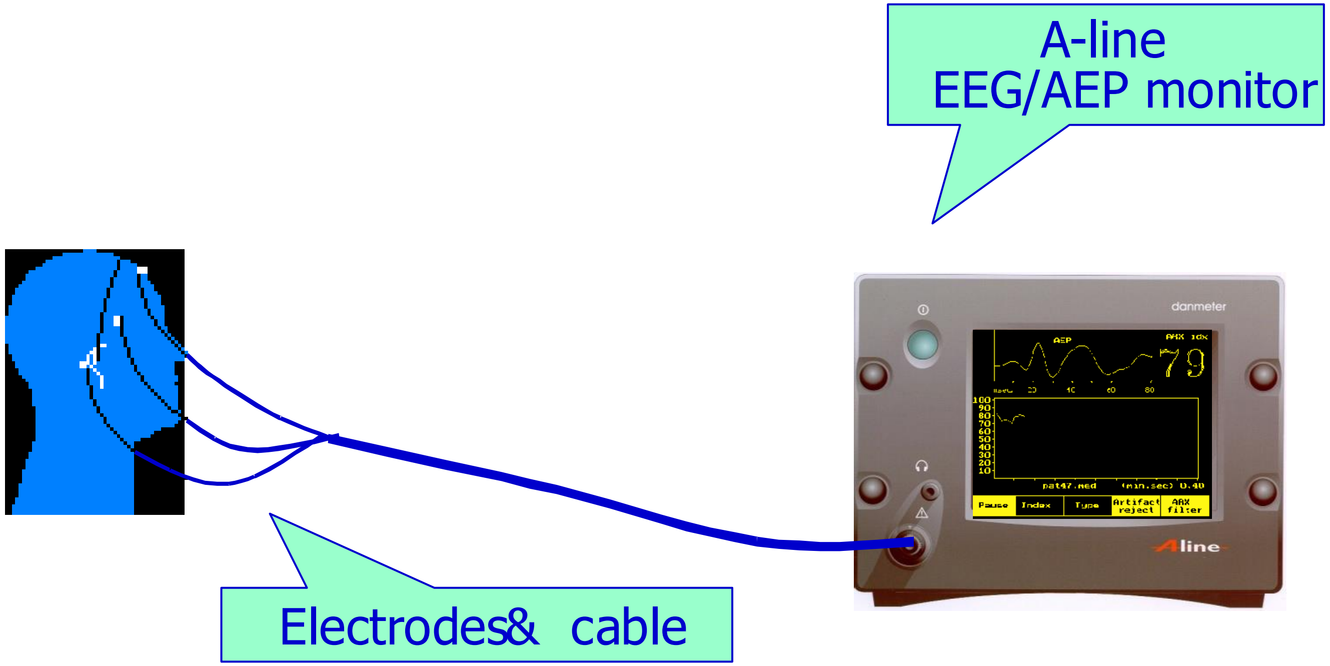
The extraction of a consistent and reliable measure online and close to real time to assess the hypnotic level during anesthesia is a continuous challenge to the anesthetist. Hemodynamic parameters such as heart rate and blood pressure are not, at least with the traditional single parameter versus time presentation, adequate for ensuring an optimal level of anesthesia, especially when using neuromuscular blocking agents. The objective of this study was to evaluate the symbolic dynamics applied to the EEG signal while awake and while asleep. Data was required from 100 patients scheduled for elective cardiac surgery. All patients were anesthetized with propofol, and no other drugs were administered during the study period. The results showed significant difference between awake and anaesthetized values, hence concluding that the complexity measure might serve as an indicator of anesthetic depth.

$$S_j = \sum_{i=1}^{M-1} \begin{cases} 0 \text{ si } \left| x_i^j - x_{i+1}^j \right| \geq a * sd \\ 1 \text{ si } \left| x_i^j - x_{i+1}^j \right| < a * sd \end{cases} \qquad j = 1, \ldots, N - M + 1$$

SD

Example: (N = 10, M = 5)

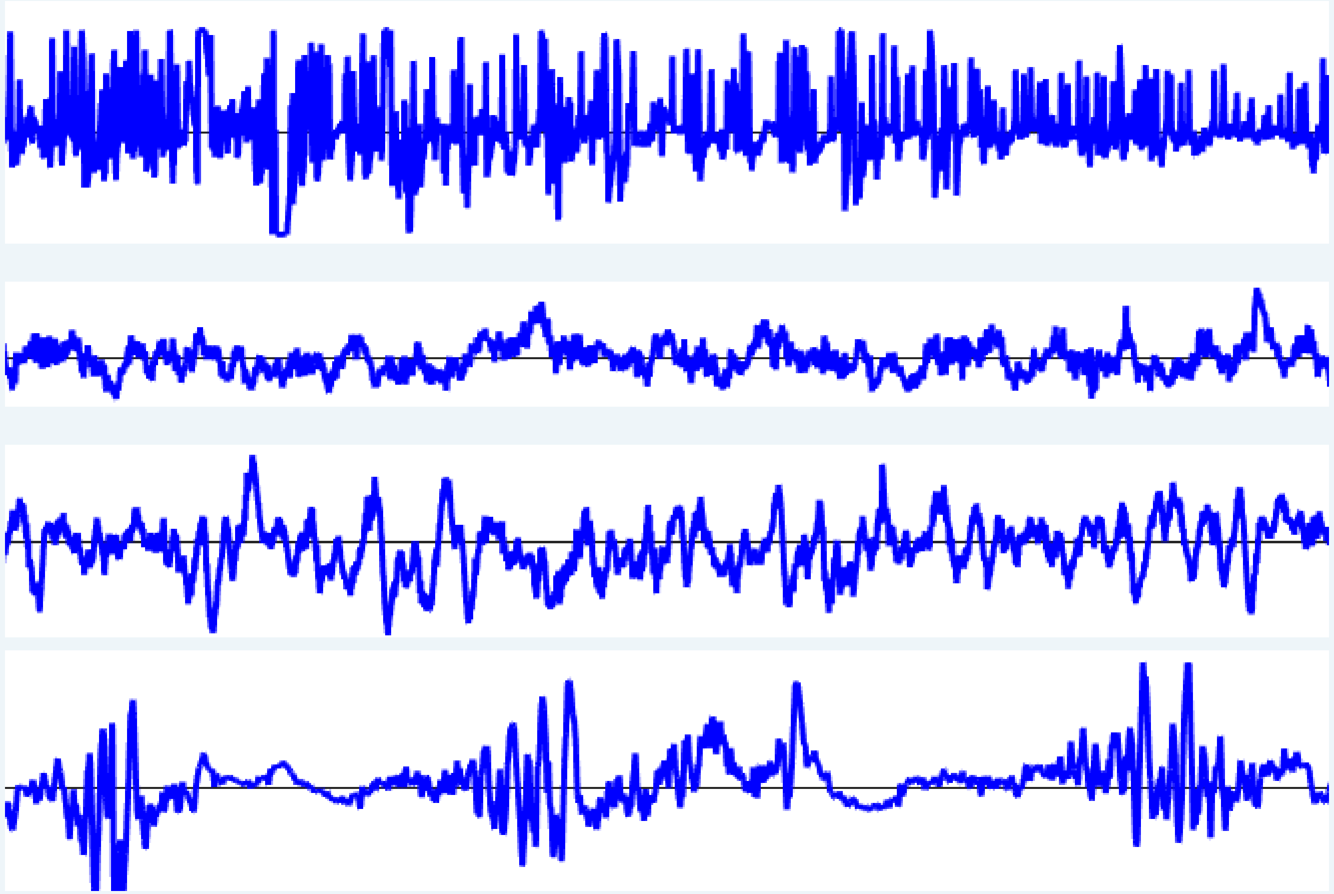



A-line
EEG/AEP monitor
Electrodes& cable
danmeter
Aline

Derived EEG

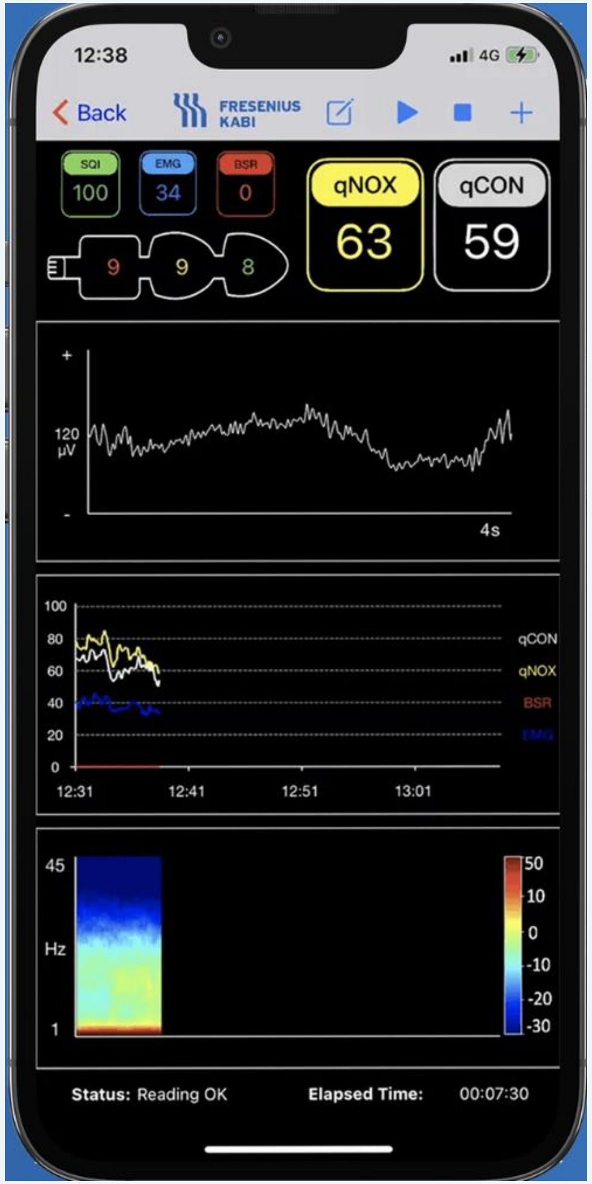
- BA: The brain activity which reflects the cortical connectivity, on a scale from 0 to 99

BA



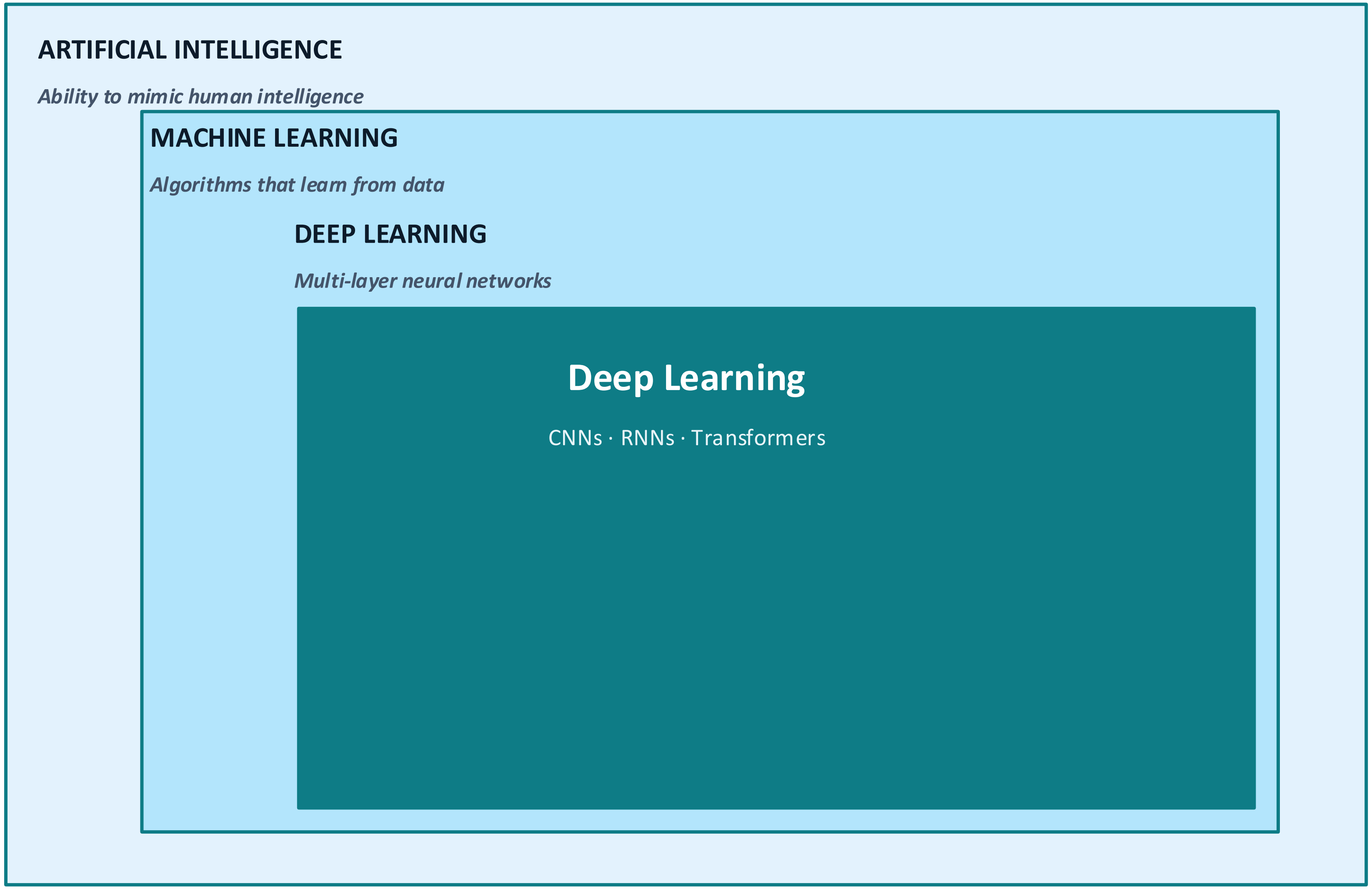
Fast waves Awake	**99**
Sedated	**70**
Anaesthetised Delta waves	**50**
Burst-suppression (BSR)	**25**


FRESENIUS KABI
SQI 100
EMG 98
BSR 0
qNOX 99
qCON
99
45
Hz
1
SEF50
SEF95
100
80
60
40
20
0
qCON
qNOX
BSR
EMG
Status:Reading OK
Elapsed time:
Time:
conox


12:38
4G
Back
FRESENIUS KABI
SQI 100
EMG 34
BSR 0
qNOX 63
qCON 59
9
9
8
+
120 µV
-
4s
100
80
60
40
20
0
12:31
12:41
12:51
13:01
qCON
qNOX
BSR
EMG
45
Hz
1
50
10
0
-10
-20
-30
Status: Reading OK
Elapsed Time: 00:07:30

The AI Hierarchy

How Machine Learning and Deep Learning fit within Artificial Intelligence



Key Concepts

AI

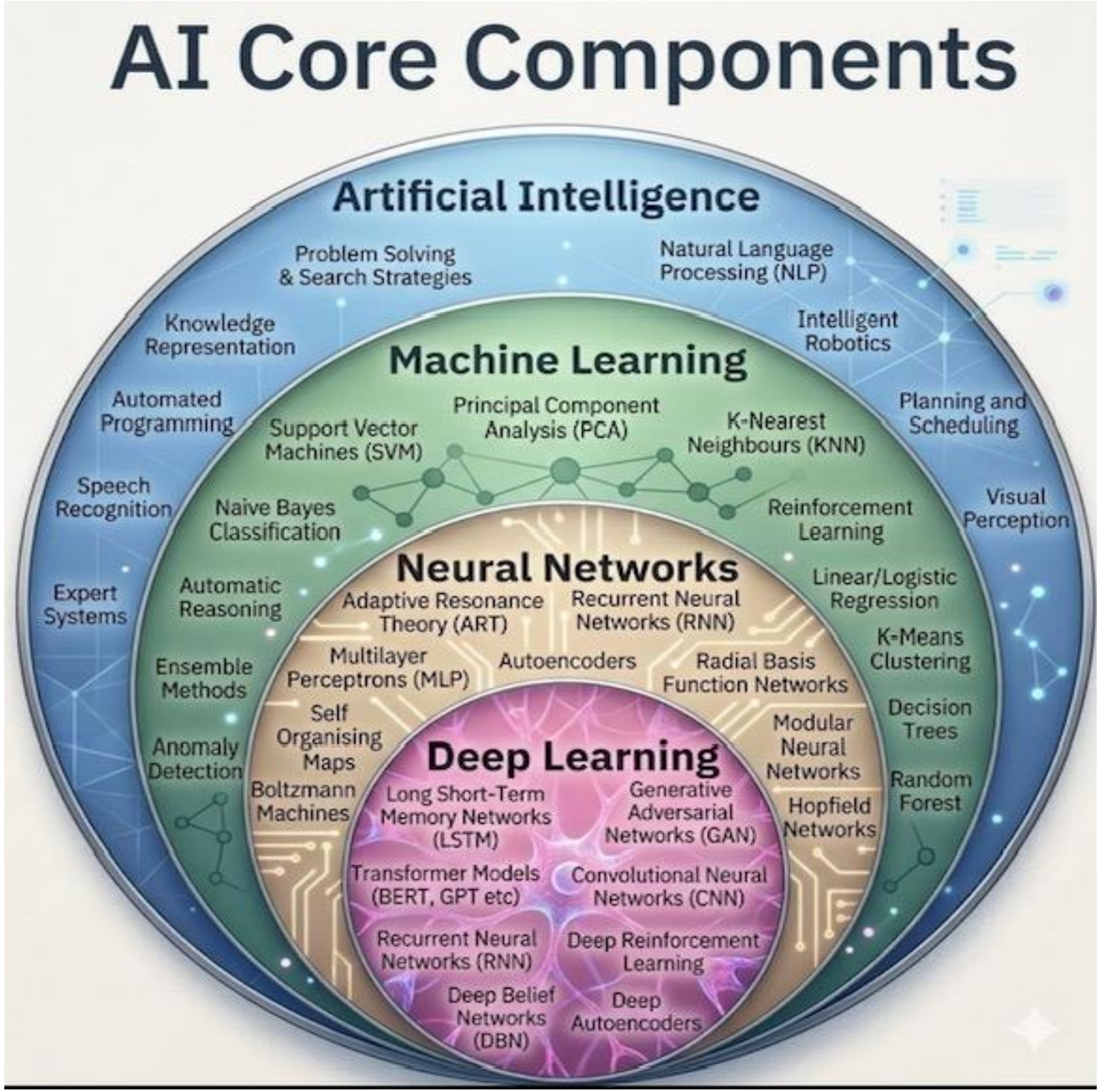
Machine that acts with human-like intelligence

ML

Computers learn from data without being explicitly programmed

DL

ML technique using multi-layer neural network architectures


AI Core Components
Artificial Intelligence
Problem Solving & Search Strategies
Natural Language Processing (NLP)
Knowledge Representation
Intelligent Robotics
Automated Programming
Planning and Scheduling
Speech Recognition
Visual Perception
Expert Systems
Machine Learning
Support Vector Machines (SVM)
Principal Component Analysis (PCA)
K-Nearest Neighbours (KNN)
Naive Bayes Classification
Reinforcement Learning
Automatic Reasoning
Linear/Logistic Regression
K-Means Clustering
Ensemble Methods
Decision Trees
Anomaly Detection
Random Forest
Neural Networks
Adaptive Resonance Theory (ART)
Recurrent Neural Networks (RNN)
Multilayer Perceptrons (MLP)
Autoencoders
Radial Basis Function Networks
Self Organising Maps
Modular Neural Networks
Boltzmann Machines
Hopfield Networks
Deep Learning
Long Short-Term Memory Networks (LSTM)
Generative Adversarial Networks (GAN)
Transformer Models (BERT, GPT etc)
Convolutional Neural Networks (CNN)
Recurrent Neural Networks (RNN)
Deep Reinforcement Learning
Deep Belief Networks (DBN)
Deep Autoencoders

Knowledge Discovery in Databases (KDD)

The six-stage process from raw data to actionable knowledge

1
Selection
Choose relevant data sources

2
Preprocessing
Handle missing values & noise

3
Transformation
Feature engineering & normalization

4
Data Mining
Apply algorithms & extract patterns

5
Interpretation
Evaluate & validate results

6
Knowledge
Deploy insights & decision support

Data Mining is the core computational step — it extracts patterns using ML algorithms. The remaining steps ensure data quality and result validity.

What is Machine Learning?

Development of computer algorithms to transform data into intelligent action

"The study of how to make computers learn or adapt — the goal is to make computers improve their performance through experience."

1 Data

Available Data

Historical records, sensor readings, images, text — the raw material that drives learning.

2 Algorithms

Statistical Methods

Mathematical procedures that detect patterns and relationships hidden within the data.

3 Computing Power

Processing

CPU/GPU capacity that scales learning to billions of examples in real time.

DM vs ML: Data Mining teaches computers to find patterns humans then interpret. ML teaches computers to use data to solve problems autonomously.

How Machines Learn — The Four Components

Every ML system is built on four inextricably linked stages

Step 1	**Data Storage** Computers store training examples in RAM. Stored data alone is insufficient — it must be processed to extract meaning.	*Key: Memorize representative ideas, not every instance.*
Step 2	**Abstraction** The algorithm builds an internal model — explicit description of patterns in data. Types: equations, rules, probability tables, clusters.	*Key: Abstraction turns raw data into a reusable structure.*
Step 3	**Generalization** The model is tested on unseen data. Bias drives action but must be balanced against variance for good generalization.	*Key: Good models handle new inputs they have never seen.*
Step 4	**Evaluation** Learner's success measured on held-out test data. Noise (measurement error, complex phenomena) is acknowledged but not modeled.	*Key: Overfitting = modeling noise → too complex a model.*

Types of Machine Learning

Three fundamental paradigms — differing in how the algorithm receives feedback

Supervised Learning

Training data includes correct output labels. The model learns to map inputs to outputs.

Classification:

Predict which discrete category an example belongs to (class label).

Regression:

Predict a continuous numeric value — essentially function approximation.

Examples: spam detection, fraud classification, house price prediction, medical diagnosis

Unsupervised Learning

No labels provided. The algorithm discovers structure in the data on its own.

Clustering:

Group similar data points together — customer segmentation, market baskets.

Pattern Discovery:

Find useful associations without a predefined target variable.

Examples: customer segments, topic modeling, anomaly detection, gene expression

Reinforcement Learning

An agent learns by interacting with an environment, receiving rewards for actions.

Reward Signal:

Algorithm maximizes cumulative reward over time through trial and error.

Applications:

Game playing, robotics, autonomous driving, recommendation systems.

Examples: AlphaGo, self-driving cars, robotic arms, dynamic pricing engines

Datasets, Features & Model Evaluation

The raw material of machine learning — structure, types, and quality assessment

Dataset Structure

Examples (Rows): Each row = one unit of observation (sample, record, instance)

Features (Cols): Recorded properties or attributes of each example

Target Variable: The outcome to be predicted (class label or numeric value)

ML Practice Pipeline

1. Data Collection
2. Data Exploration & Preparation
3. Model Training
4. Model Evaluation
5. Model Improvement & Deployment

Feature Types

- **Numeric:** Continuous/discrete numbers — age, temperature, price
- **Categorical:** Unordered categories — color, city, blood type
- **Ordinal:** Ordered categories — small/medium/large, rating 1–5
- **Non-ordinal:** Categories without natural order or scale

Overfitting vs. Underfitting

Overfitting: Model memorizes noise → great on training data, poor on new data. Too complex.

Underfitting: Model too simple → high error on both training and test sets.

Remedy: Cross-validation, regularisation, pruning, ensemble methods.

Prediction — Core Concept & Workflow

Assigning an input to one of N predefined categories using a learned model

TWO phases: (1) Learning Phase — train the model (2) Evaluation Phase — test on new data

Phase 1 — Learning (Training)

- **Understand Problem:** Identify potential features and the label to be predicted.
- **Collect Labeled Data:** Gather examples where each input has a known correct output class.
- **Select Algorithm:** Choose a classifier (kNN, Naïve Bayes, Decision Tree, SVM, etc.).
- **Train the Model:** Feed labeled training data — model learns decision boundaries.
- **Store the Model:** Save learned parameters for prediction on unseen inputs.

Phase 2 — Evaluation (Testing)

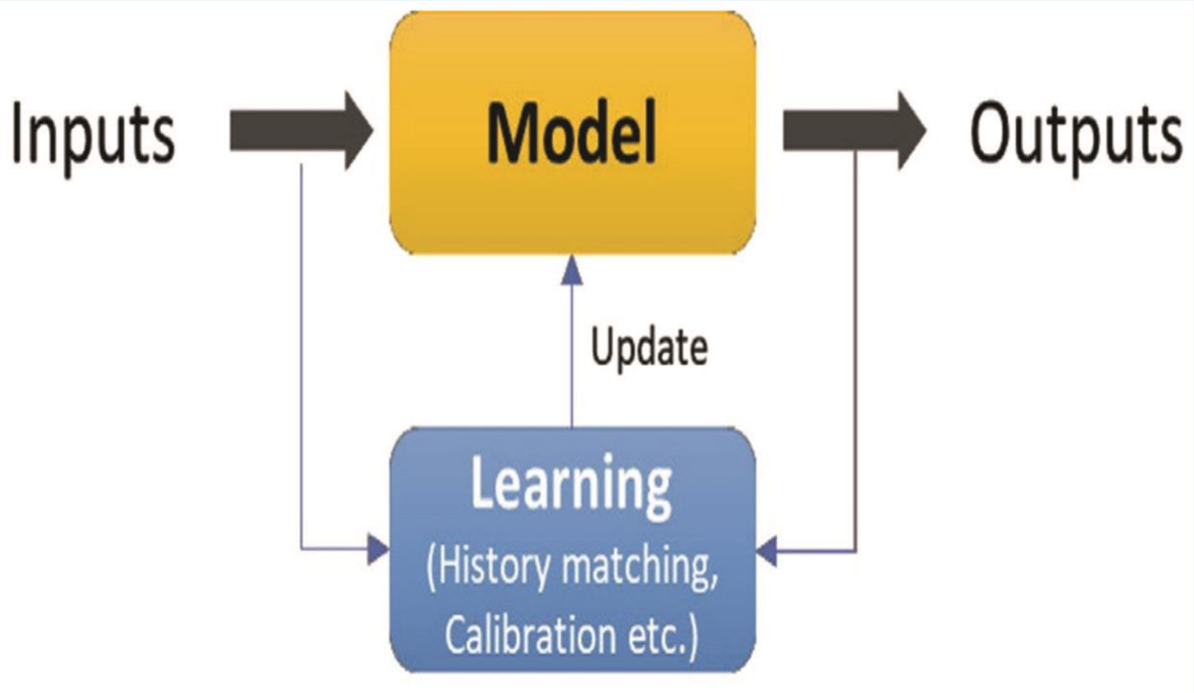
- **Apply to Test Data:** Run model on held-out set the model has never seen.
- **Compare Predictions:** Check predicted labels against known true labels.
- **Compute Metrics:** Calculate Accuracy, Precision, Recall, F1-score, AUC.
- **Detect Overfitting:** Train acc >> Test acc → overfitting; simplify or regularize.
- **Iterate:** Use results to re-tune hyperparameters and improve model.

Applications

In order to better model complex world real-data from different fields, including geosciences and medical sciences, one approach is to develop:

- Pattern recognition techniques
- Robust features
- Data Mining
- Machine learning
- Deep learning methods
- Predictive analytics

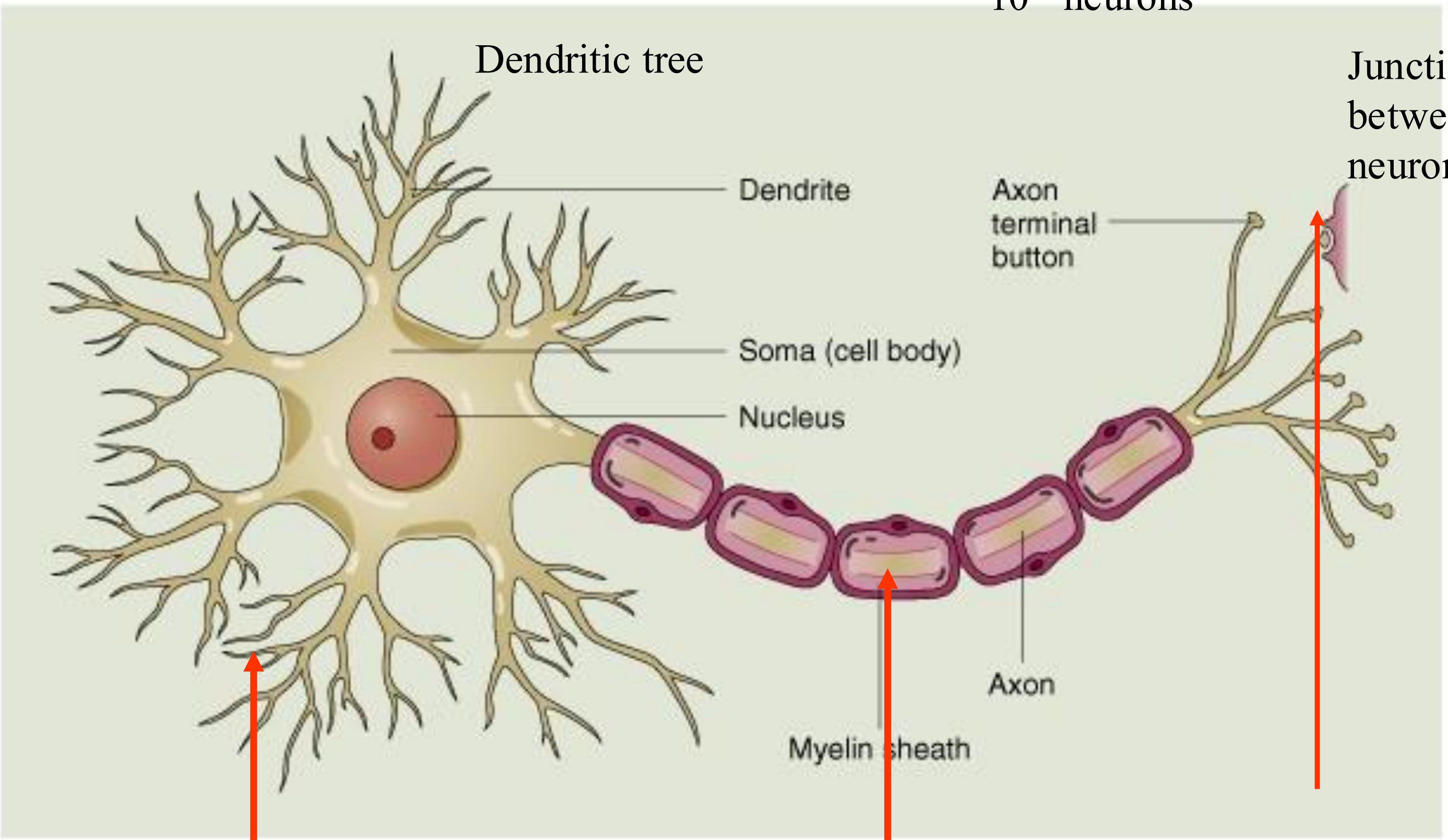
Training


Inputs
Model
Outputs
Update
Learning
(History matching,
Calibration etc.)

Deep Learning

A Typical Cortical Neuron

10^{11} neurons



© 2000 John Wiley & Sons, Inc.

Collect chemical signals

Axon: generate
Potentials (Fire/not Fire)

Synapse: control
release chemical
transmitters.

From Biological to Artificial Neurons

ANNs model the structure and function of the human brain

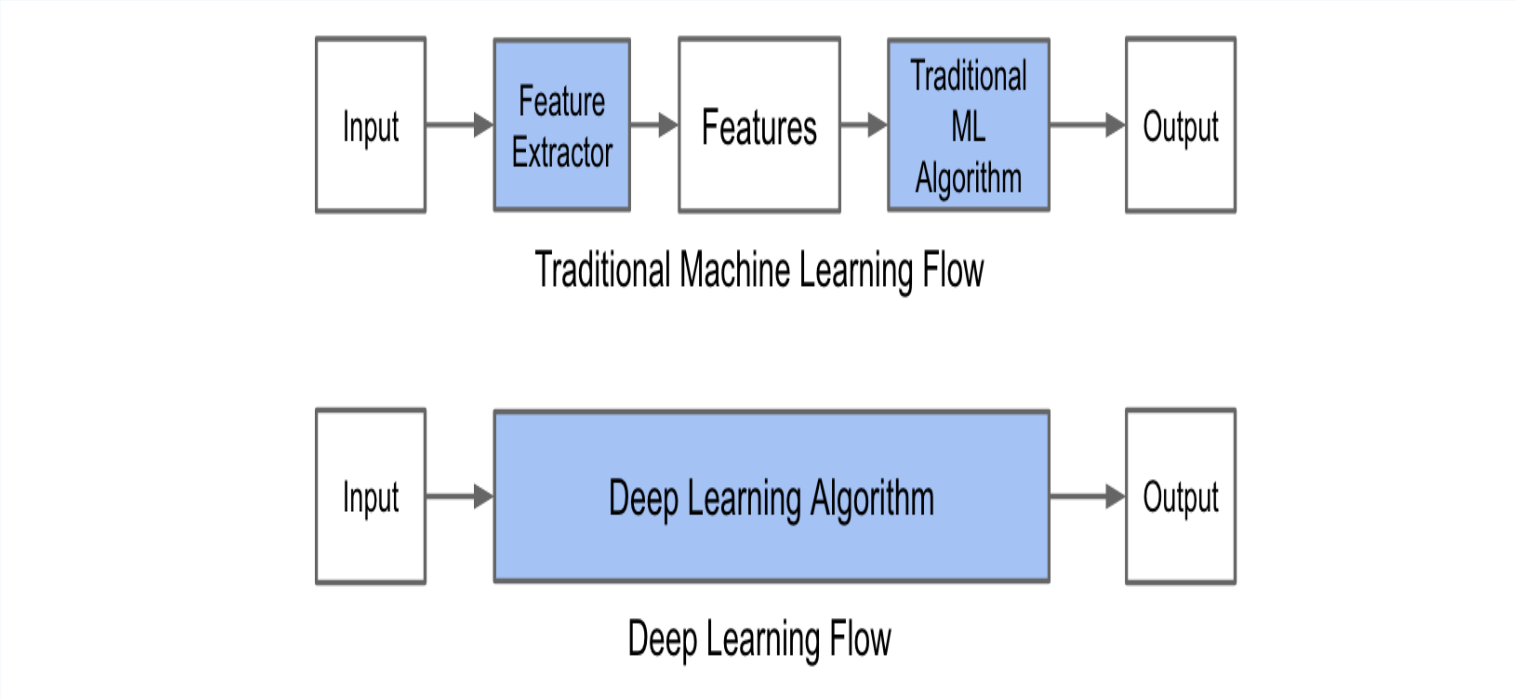
Biological Neuron

- **Dendrites:** Receive incoming signals from other neurons
- **Cell Body:** Sums up incoming signals — integration center
- **Threshold:** Fires when accumulated signal exceeds threshold
- **Axon:** Transmits output signal to neighboring neurons
- **Synapse:** Gap where electrochemical signal passes between neurons

Artificial Neuron (ANN Node)

- **Input x:** Feature values / signals from previous layer
- **Weights w:** Importance multipliers — learned during training
- **Summation Σ:** Weighted sum: $z = w^T x + b$ (bias b adjusts threshold)
- **Activation f(z):** Non-linear function: ReLU, Sigmoid, Tanh, Softmax
- **Output y:** Signal passed to next layer or final prediction

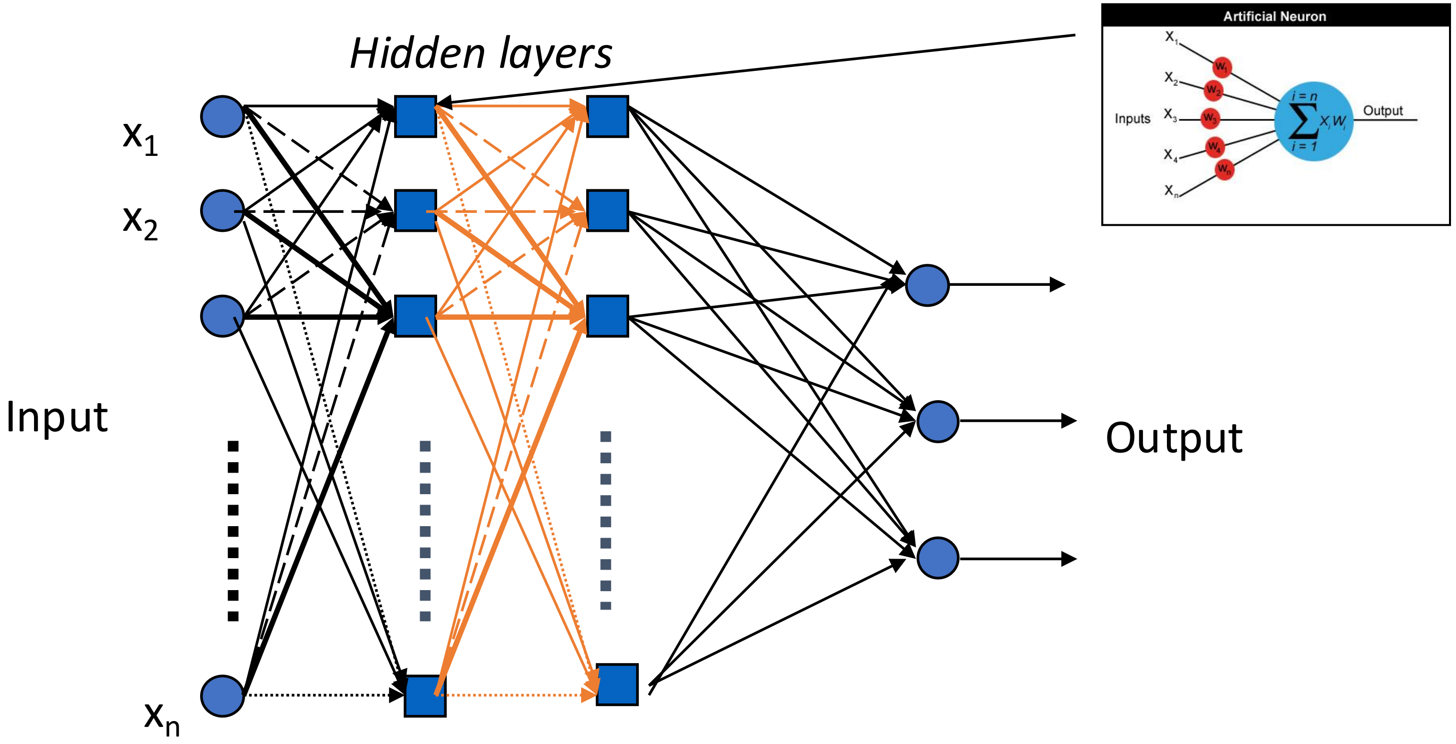
Deep Learning - Neural Network



The main difference between traditional machine learning and deep learning algorithms is in the feature engineering. In traditional machine learning algorithms, we need to hand-craft the features. By contrast, in deep learning algorithms feature engineering is done automatically by the algorithm. Feature engineering is difficult, time-consuming and requires domain expertise. The promise of deep learning is more accurate machine learning algorithms compared to traditional machine learning with less or no feature engineering.

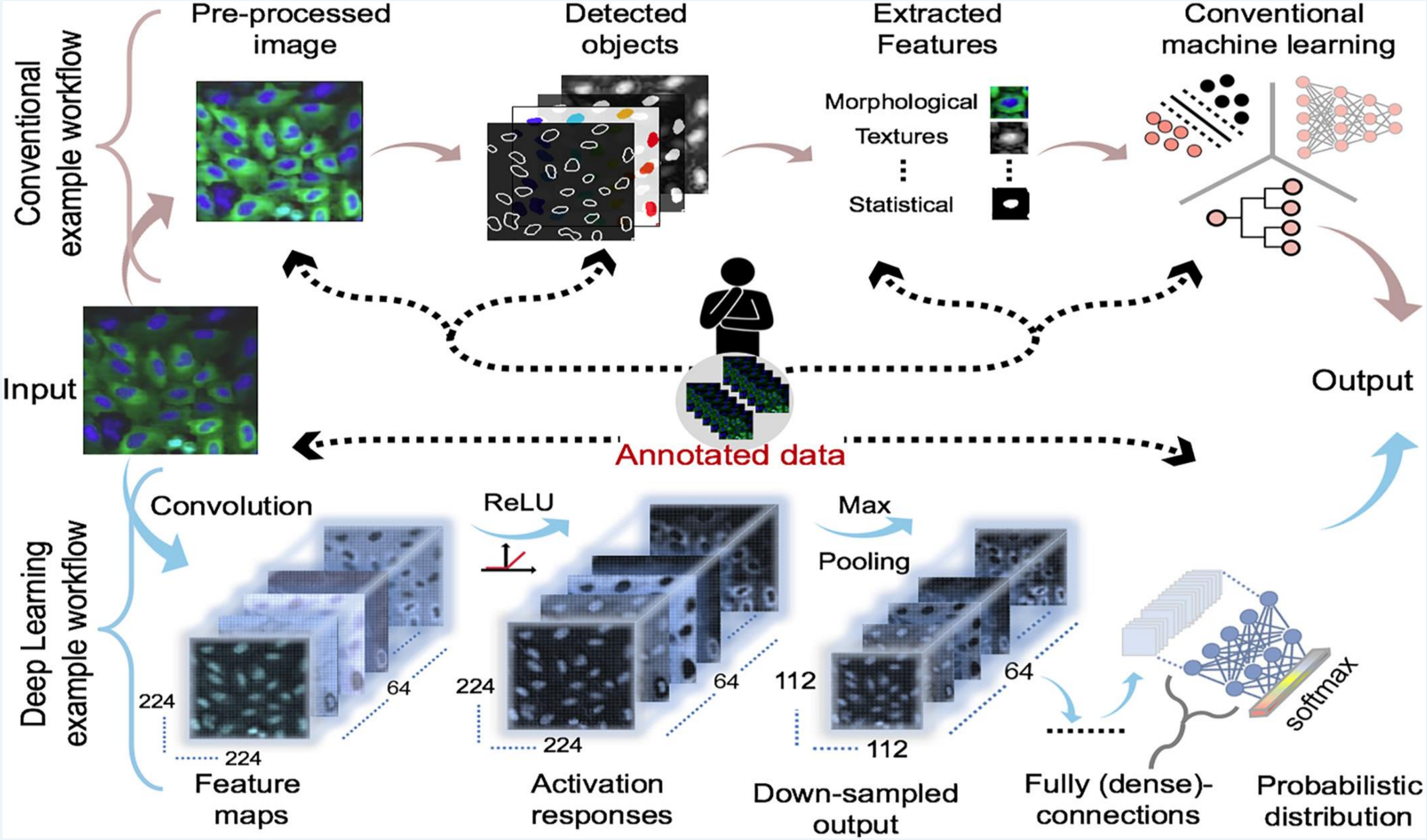
Inside a Deep Neural Network

- A deep neural network combines multiple nonlinear processing layers, using simple elements operating in parallel and inspired by biological nervous systems. It consists of an input layer, several hidden layers, and an output layer. The layers are interconnected via nodes, or neurons, with each hidden layer using the output of the previous layer as its input.

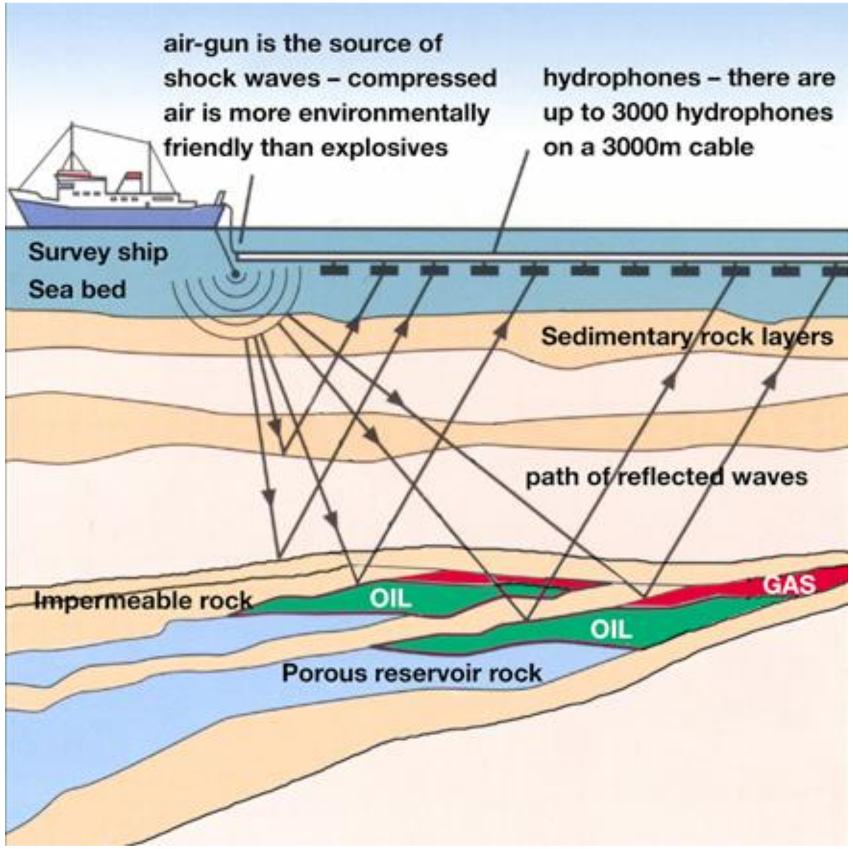


Deep learning neural networks require large amounts of training data to learn and extract features from big data sets.

Deep Learning - Neural Network



Seismic Acquisition and Database



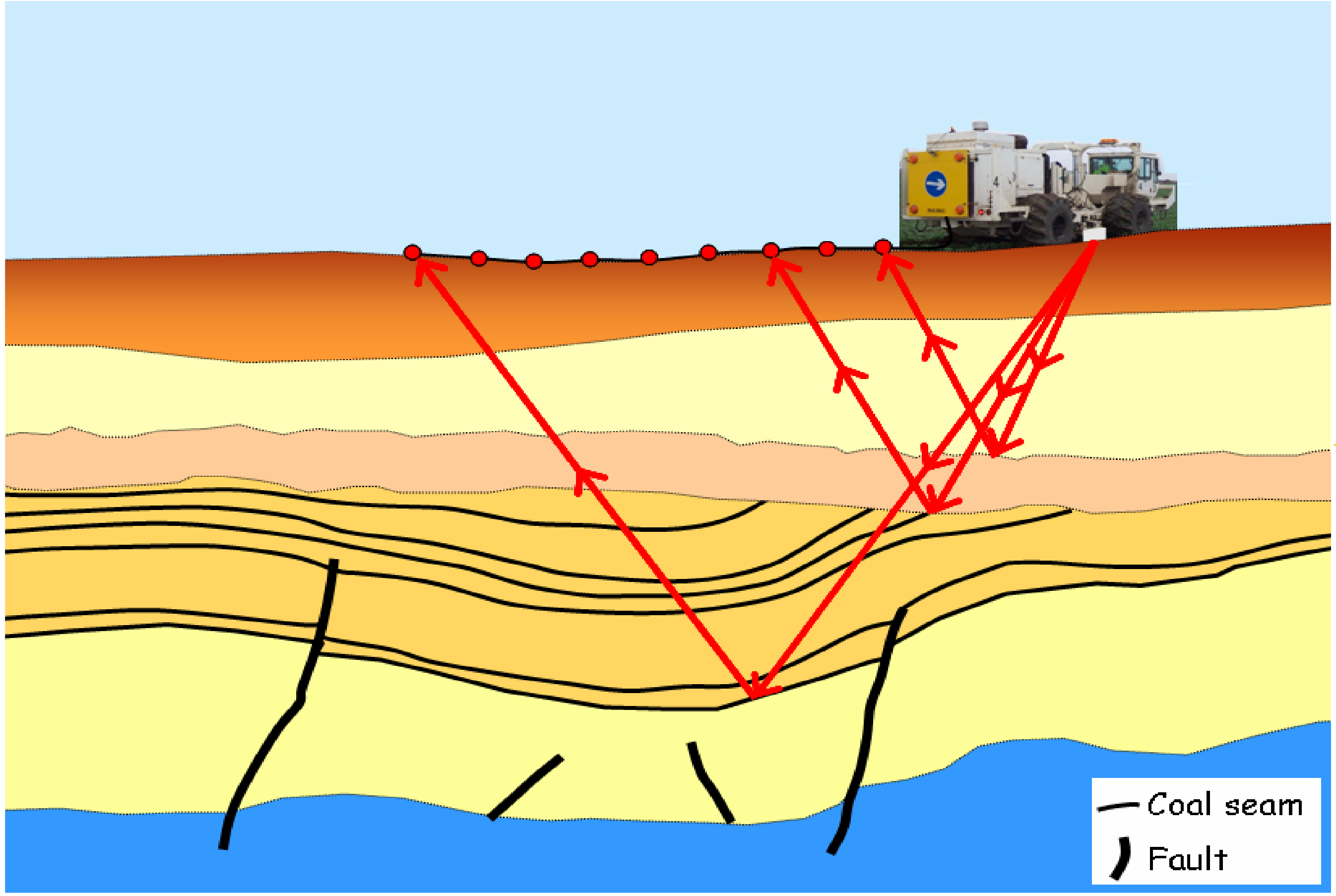
Seismic acquisition offshore:

- An air gun towed behind the survey ship transmits sound waves through the water column and into the subsurface
- Changes in rock type or fluid content reflect the sound waves towards the surface
- Receivers towed behind the vessel record how long it takes for the sound waves to return to the surface
- Sound waves reflected by different boundaries arrive at different times.
- The same principles apply to onshore acquisition

Seismic Acquisition and Database

Seismic acquisition Onshore:

- **Onshore seismic acquisition requires an energy input from a "thumper" truck. Geophones arrayed in a line behind the truck record the returning seismic signal**



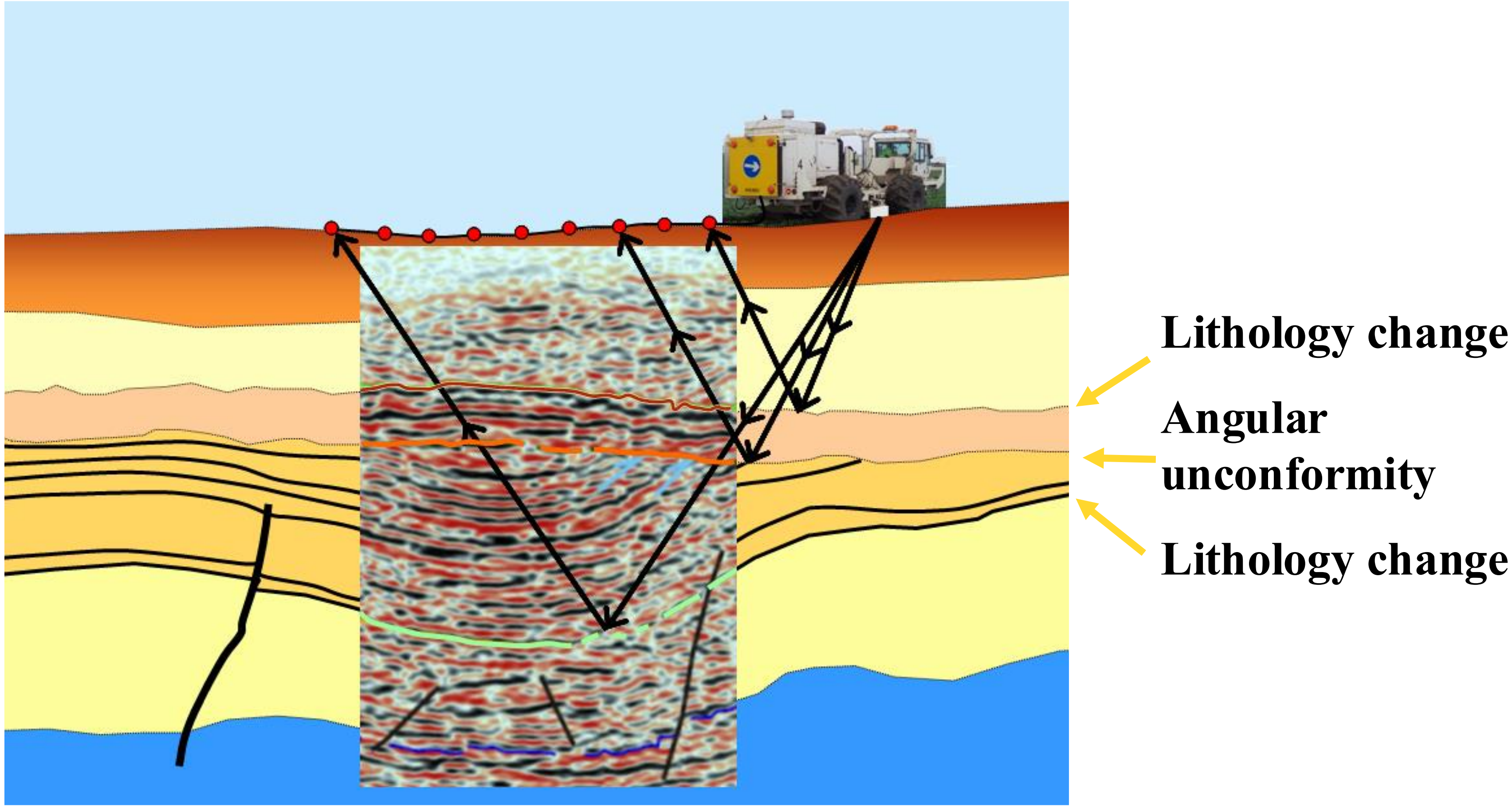
Sub-horizontal beds

Unconformity

Dipping beds

Seismic Acquisition and Database

- **Seismic horizons represent changes in density and allow the subsurface geology to be interpreted.**

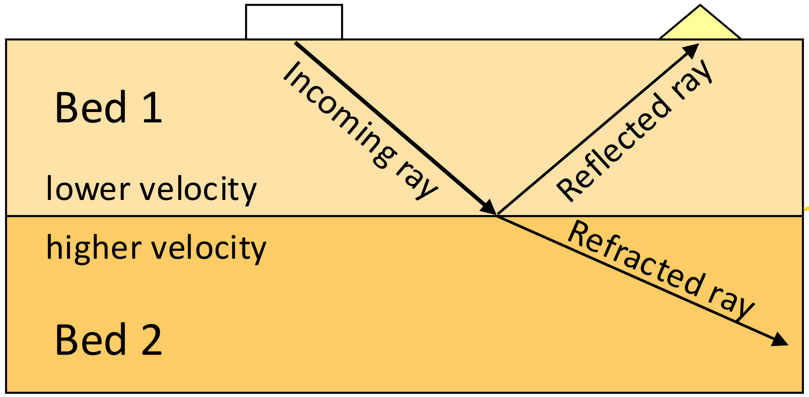


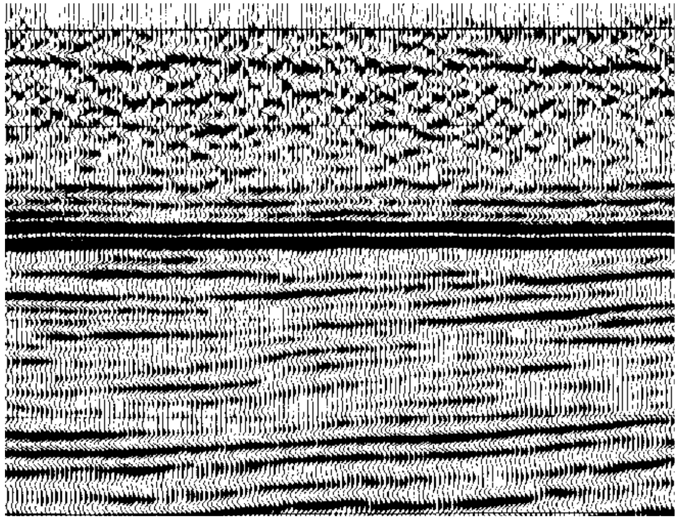
Seismic Acquisition and Database

What is a reflector?

A seismic reflector is a boundary between beds with different properties. There may be a change of lithology or fluid fill from Bed 1 to Bed 2. These property changes cause some sound waves to be reflected towards the surface.

There are many reflectors on a seismic section. Major changes in properties usually produce strong, continuous reflectors as shown by the arrow.





Seismic Acquisition and Database

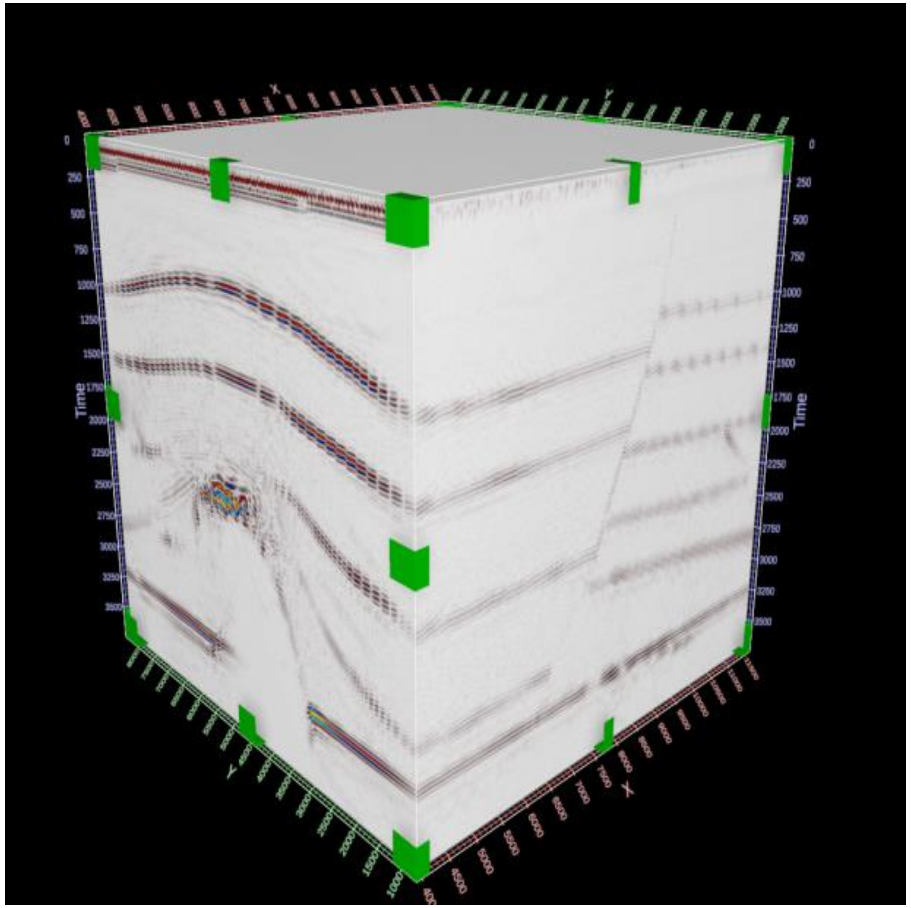


Figure. Seismic data cube

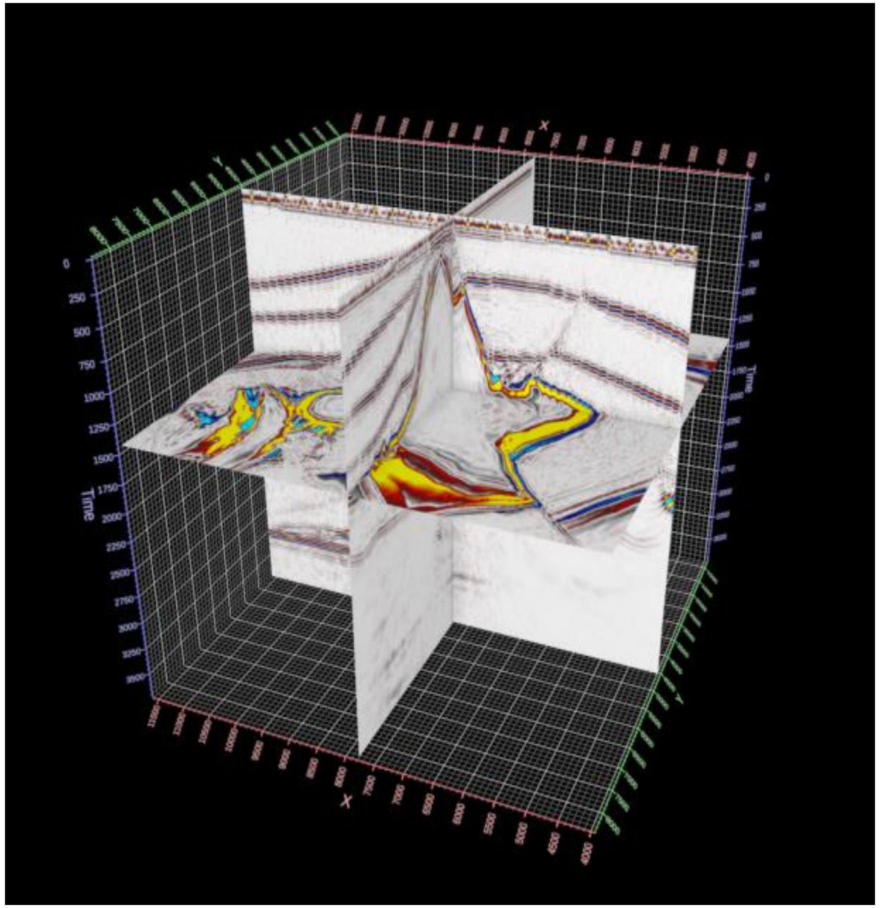
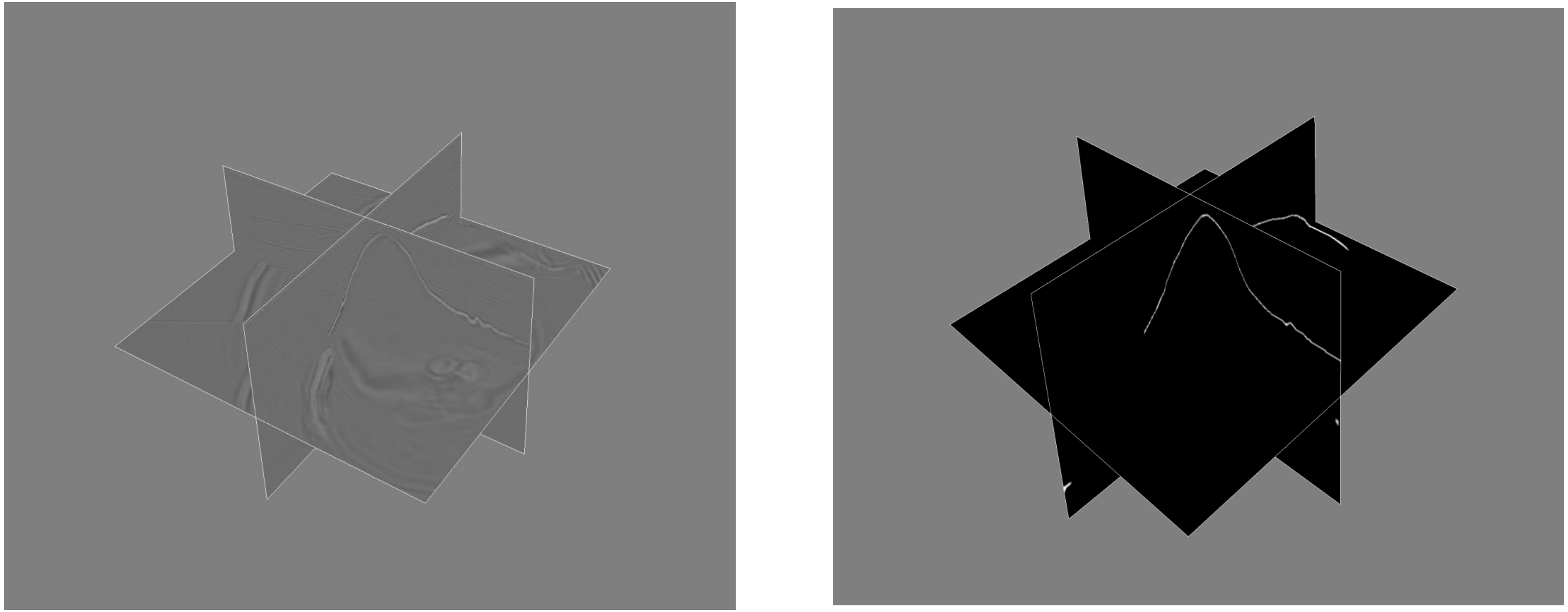


Figure. 3D dataset showing a specific in-line, cross-line and time slide



(a) *(b)*

Fig. 2. (a) Seismic data, (b) Salt top

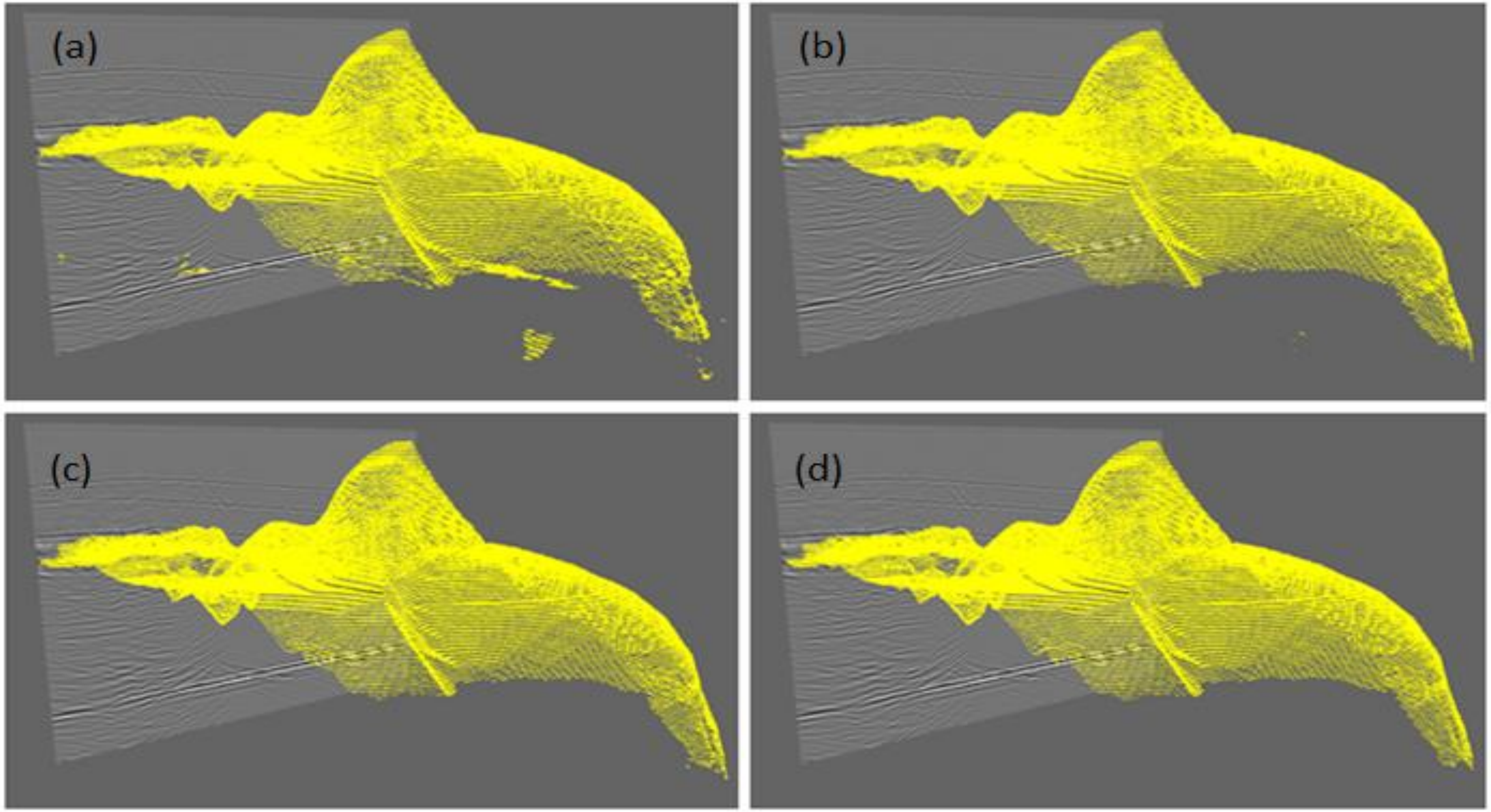


Fig. 3. Salt top extraction for different epochs. (a) epoch = 5, (b) epoch = 20, (c) epoch = 40, (d) epoch = 80

Experimental Results

TABLE 2. ACCURACY FOR TESTING DATA

KNN	*68%*
SVM	*92 %*
RF	*92 %*
CNN	*96 %*

CNN can extract useful and hidden features during training automatically from the original images, which is evidenced by the results obtained in this study.

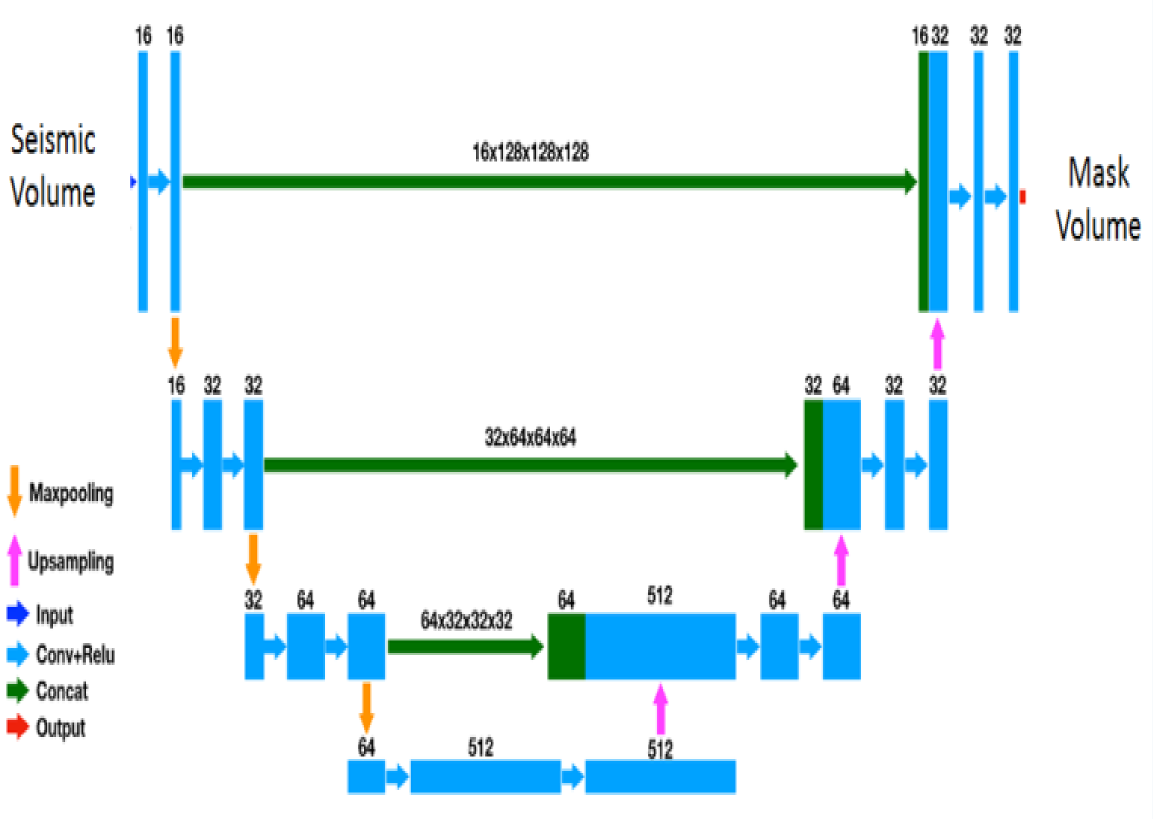


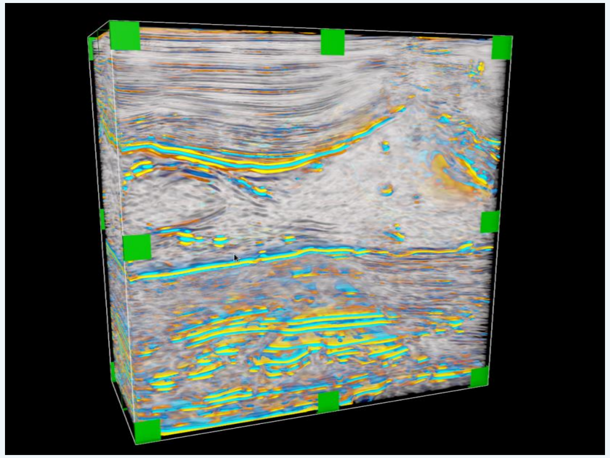
Fig. U-NET architecture

- Segmentation task using a convolutional deep learning platform to predict fault in a seismic image.
- The output of the neural networks is a mask image with the same size as the input image.
- Every voxel in the input image is classified, often by a binary label such as 0 for no fault and 1 for fault.

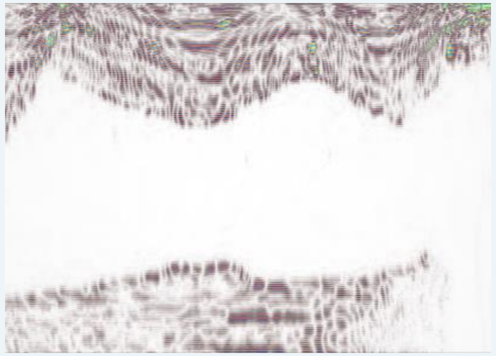
Where can machine learning be applied?
Application 1

Learning to classify salt structures in seismic data for reservoir (oil/gas) characterization

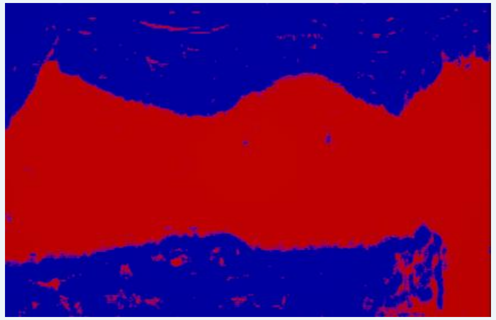
- 3D Seismic data



- Features

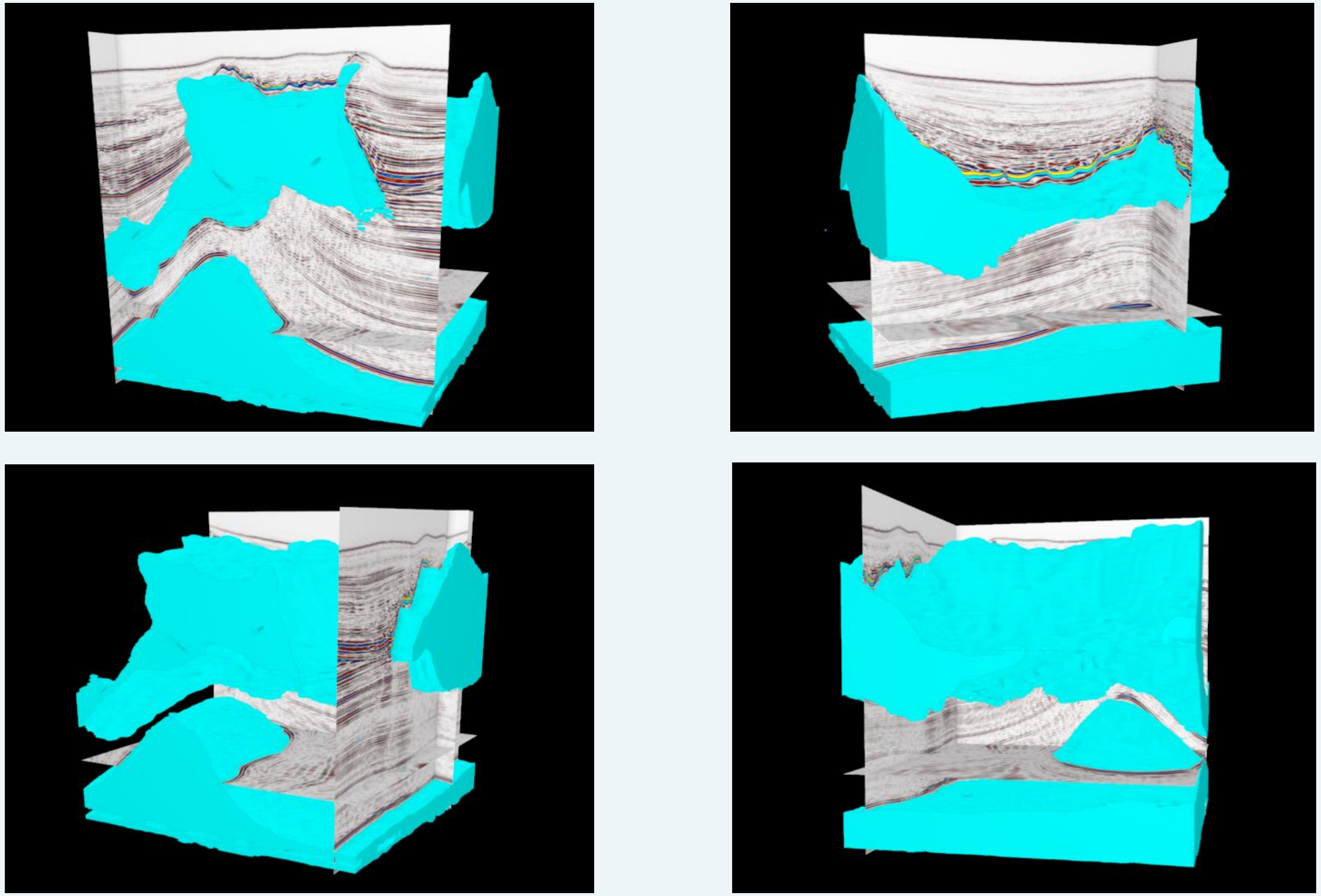


Machine learning (0 salt, red, 1 sediments, blue)

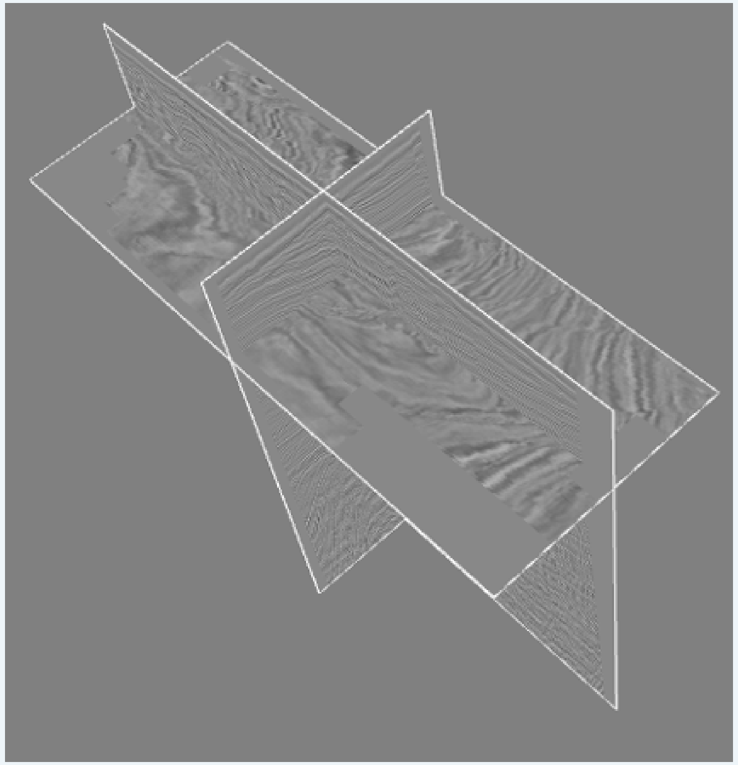


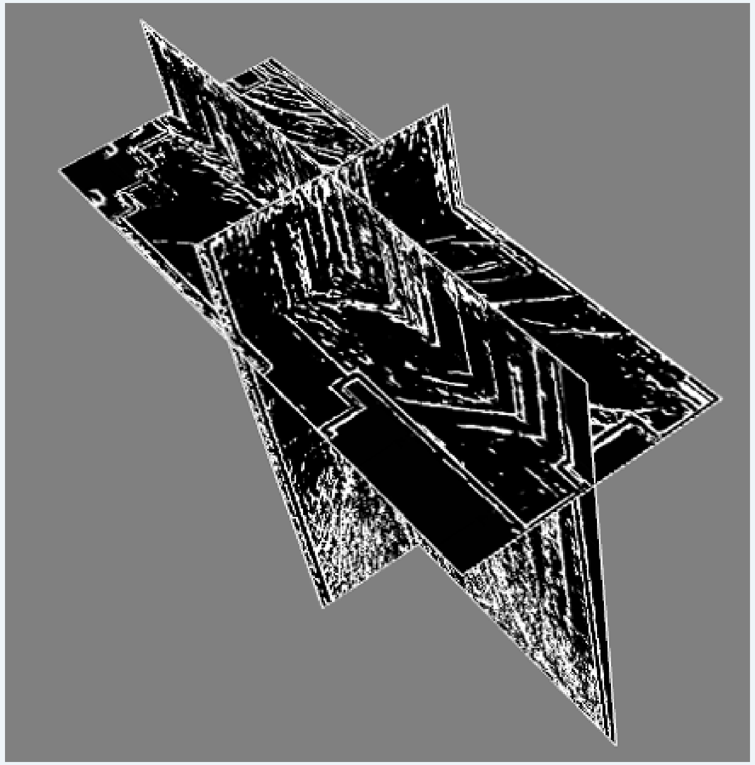
Where can machine learning be applied? Application 2

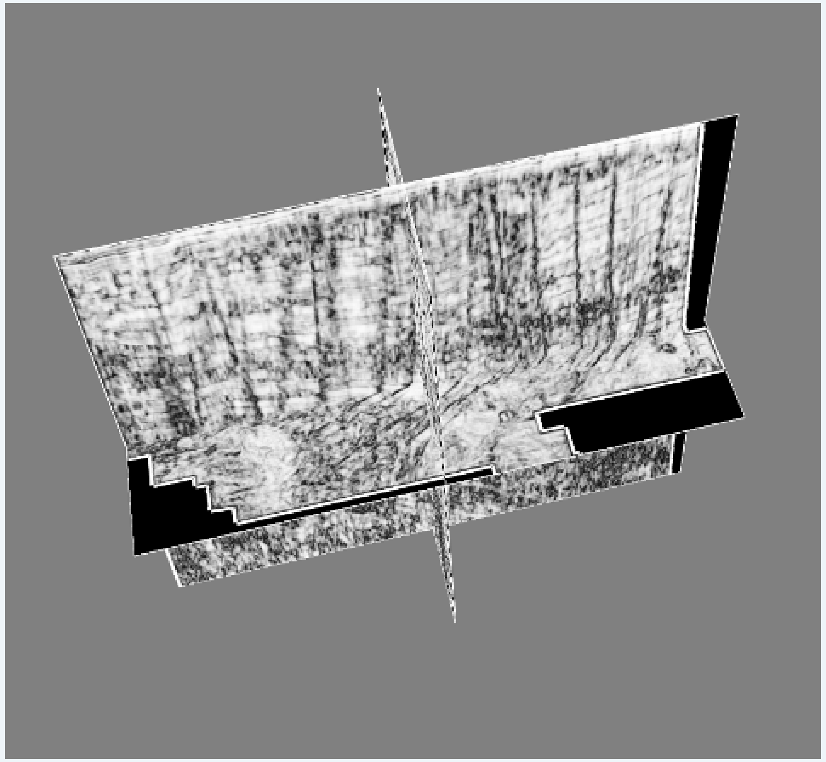
Geobody Detection



Where can machine learning be applied?
Application 3





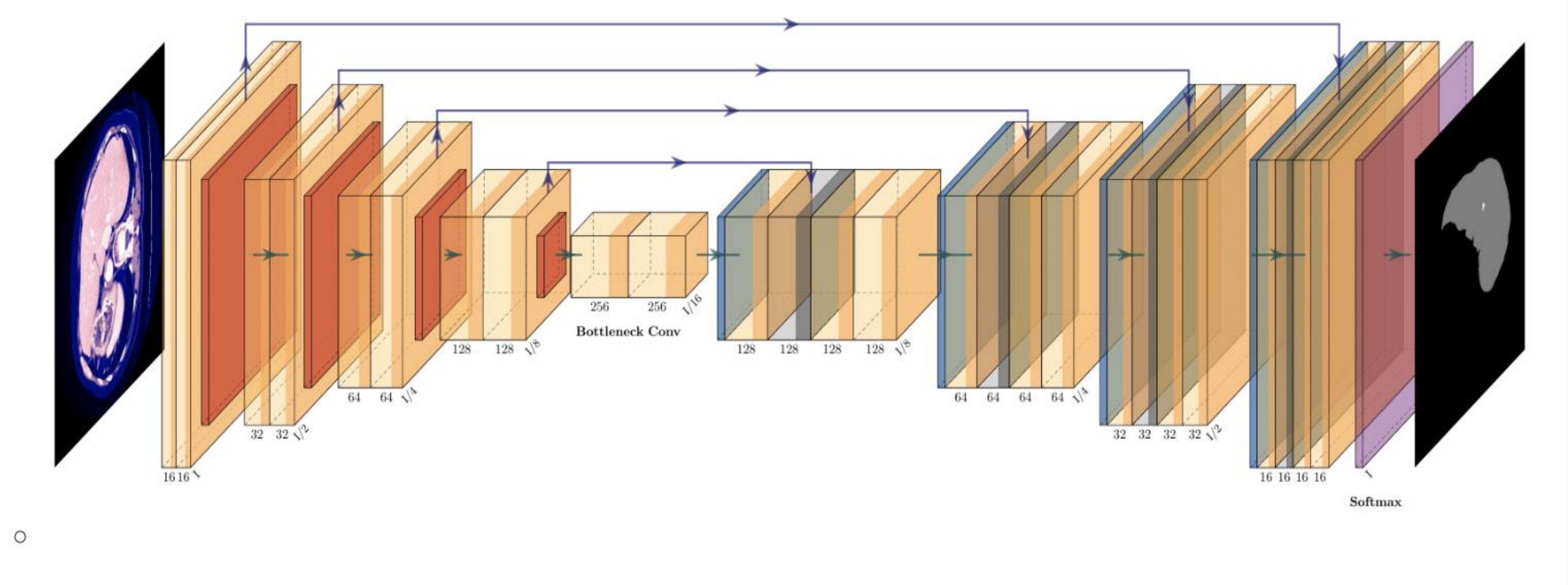


Seismic

Fault detection

Where can machine learning be applied?
Application 4

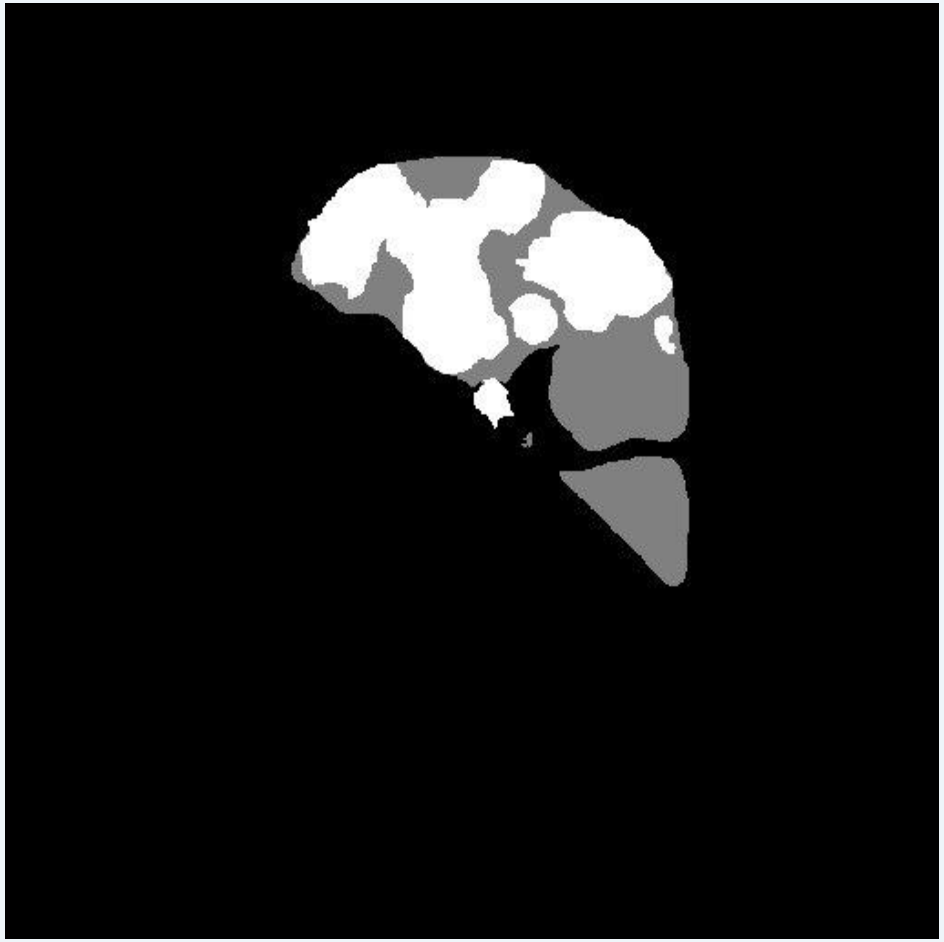
Multiclass U-Net Segmentation for Liver Tumor Detection in CT-Scan Images



Where can machine learning be applied?
Application 4 (Cont)

Multiclass Semantic Segmentation:

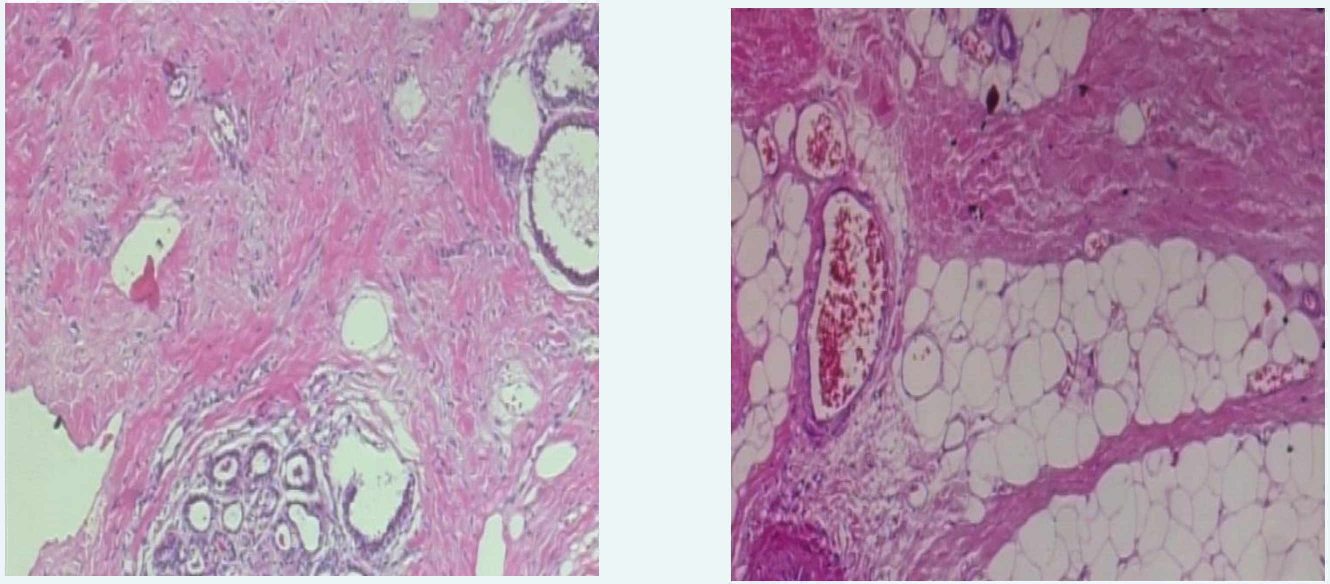
Unlike traditional binary segmentation, this approach supports multiclass segmentation. This means the model can distinguish between different classes of tissues, allowing for more nuanced and detailed segmentation, crucial for accurate liver tumor detection.



Where can machine learning be applied?
Application 5

Deep Learning

Breast Cancer Classification: A Deep Learning Approach for Digital Pathology



(a) (b)

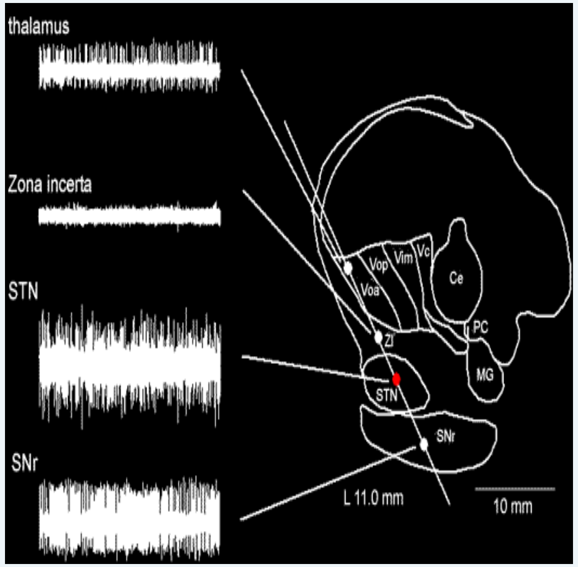
Fig. 1. (a) A slide of breast benign tumor, and (b) a slide of breast malignant tumor.

Accuracy of 91 %

Where can machine learning be applied?
Application 6

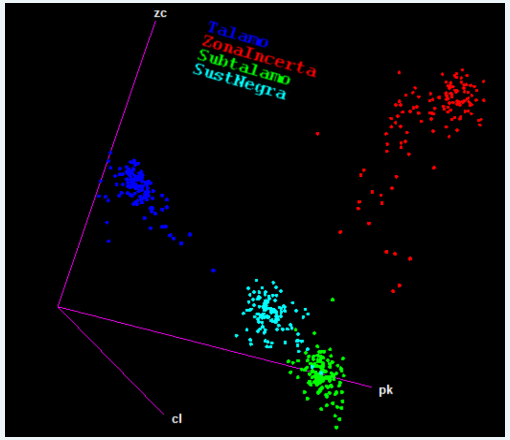
Deep brain stimulation for Parkinson's disease

Learning to classify subcorticals structures for DBS



- Features

Machine learning



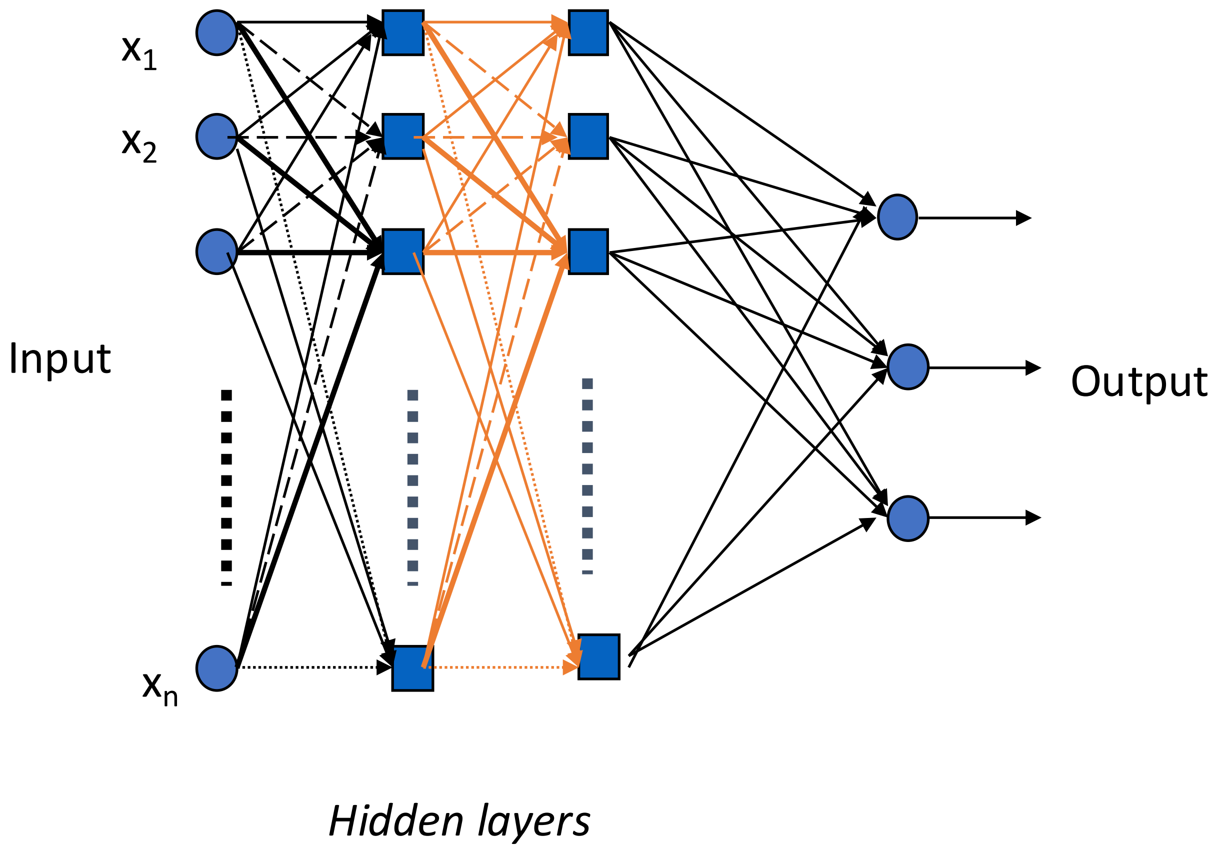
Neural Networks:

$$\hat{y} = \Sigma\, f(m_i x_i + b)$$

(applied across hundreds or billions of neurons, in many layers)

m	**Weights (m):** Learned coefficients — billions of them
x	**Inputs (x):** Pixels, words, sensor readings…
f	**Activation f(·):** Adds non-linearity — the secret sauce
Σ	**Sum + Stack:** Layer upon layer of mx + b

Three-layer networks



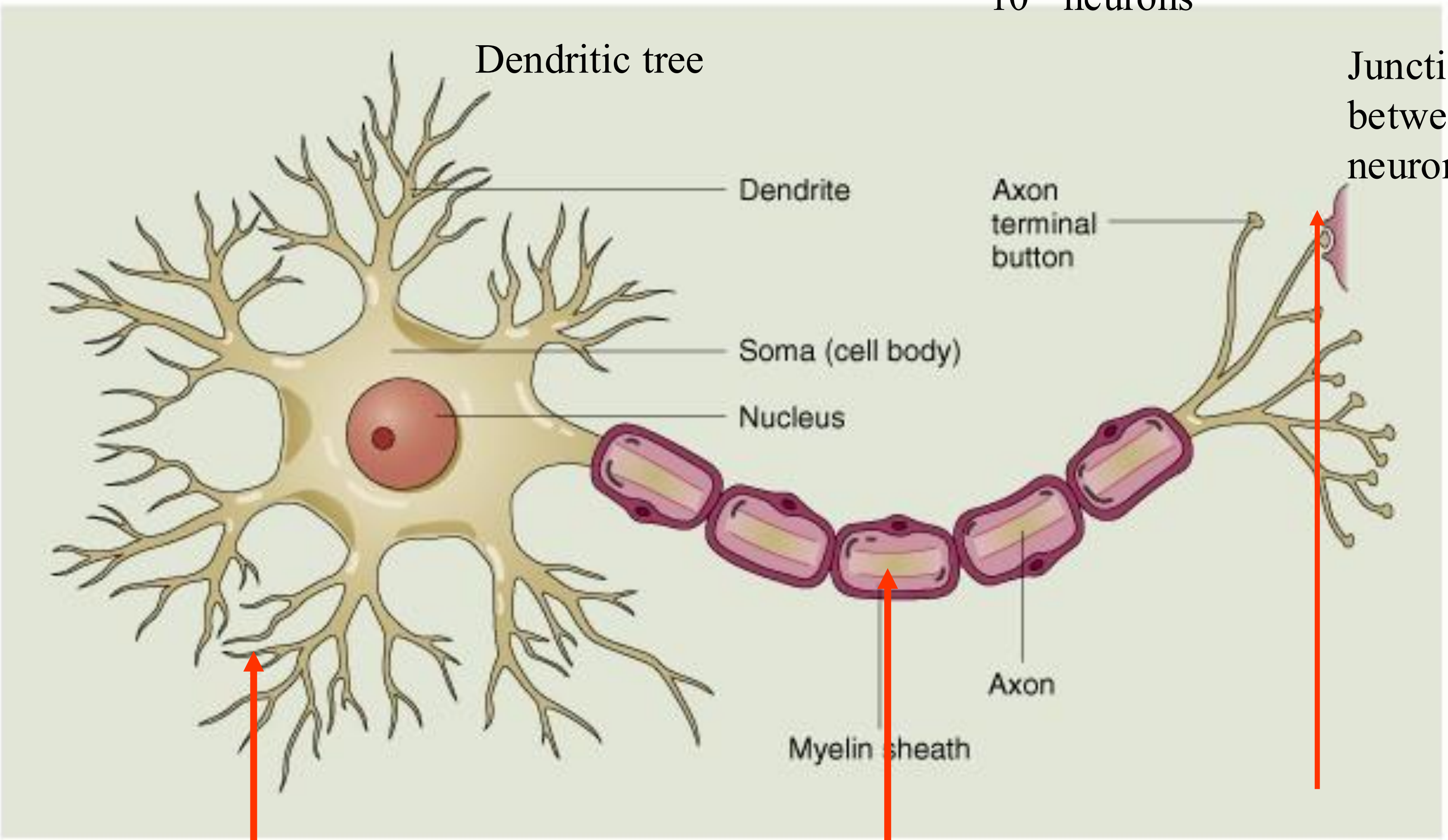
The math hasn't changed — only the scale of imagination has.

A Typical Cortical Neuron

10^{11} neurons

Dendritic tree

Junction between neurons



© 2000 John Wiley & Sons, Inc.

Collect chemical signals

Axon: generate Potentials (Fire/not Fire)

Synapse: control release chemical transmitters.

From Biological to Artificial Neurons

ANNs model the structure and function of the human brain

Biological Neuron

- **Dendrites:** Receive incoming signals from other neurons
- **Cell Body:** Sums up incoming signals — integration center
- **Threshold:** Fires when accumulated signal exceeds threshold
- **Axon:** Transmits output signal to neighboring neurons
- **Synapse:** Gap where electrochemical signal passes between neurons

Artificial Neuron (ANN Node)

- **Input x:** Feature values / signals from previous layer
- **Weights w:** Importance multipliers — learned during training
- **Summation Σ:** Weighted sum: $z = w^T x + b$ (bias b adjusts threshold)
- **Activation f(z):** Non-linear function: ReLU, Sigmoid, Tanh, Softmax
- **Output y:** Signal passed to next layer or final prediction

Backpropagation & The Deep Learning Revolution

From Forgotten Algorithm to World-Changing Technology

1986

Backprop Popularized

Rumelhart, Hinton & Williams publish landmark paper in Nature demonstrating that backpropagation can train multi-layer networks — reviving neural net research.

2006

Deep Belief Networks

Hinton & Salakhutdinov show deep networks can be trained layer-by-layer. Published in Science — the shot heard around AI.

2012

AlexNet & ImageNet

Hinton's group wins ImageNet with 10.8% error gap. Triggered the explosion of deep learning in academia and industry globally.

Key insight: Hinton showed that hierarchical representations — learned automatically from data via gradient descent — outperform all hand-crafted features. This principle now underlies LLMs, vision models, and generative AI.

Legacy & Continuing Revolution

How Hinton's Ideas Reshaped Computation, Science, and Society

Foundation of Modern AI

Every major AI system — GPT, Gemini, DALL·E, AlphaFold — descends from Hinton's backprop + deep network ideas.

Redistribution of Intelligence

Deep learning moved AI from narrow expert systems to general-purpose models capable of language, vision, protein folding, and more.

Safety & Existential Concern

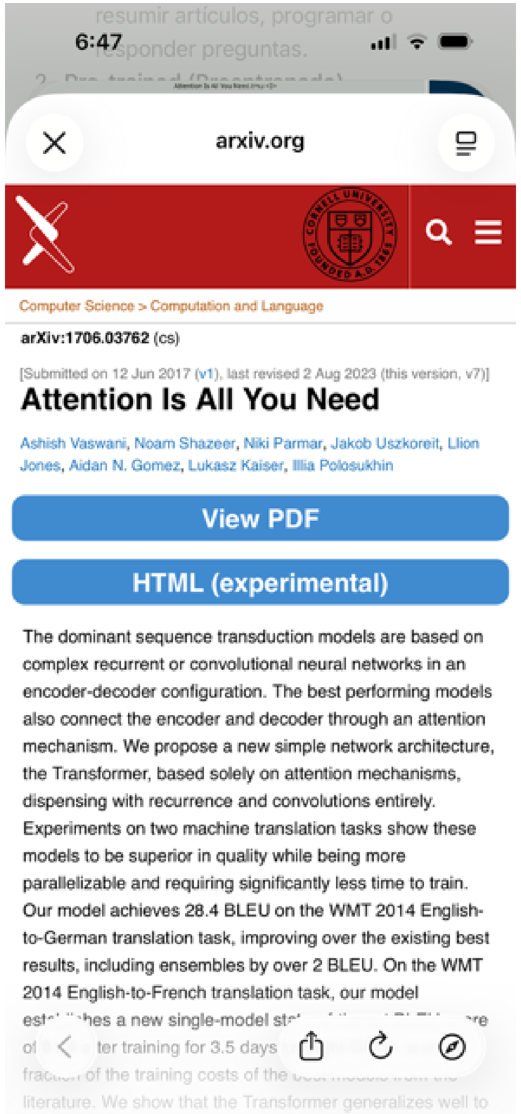
In 2023 Hinton left Google to speak freely about AI risks, warning the world about autonomous decision-making and misuse potential.

Scientific Breakthroughs

AlphaFold (proteins), drug discovery, climate modeling — deep nets now accelerate discovery across every scientific domain.

"I am now more worried about the risks of AI than I was before." — Geoffrey Hinton, 2023

Attention is All You Need



While attention-based Transformers have revolutionized machine learning, a fundamental scientific question remains open: does there exist a richer mathematical architecture, rooted in operator theory, reproducing kernels, Sobolev spaces, or functional analysis, that can provide greater expressiveness, generalization, and computational efficiency than attention itself?

Computing Neural Network Gradients

A Vectorized Approach

WHY VECTORIZED GRADIENTS?

Scale

Neural networks contain millions or billions of parameters — computing one at a time is impractical.

Speed

Matrix operations enable highly optimized parallel execution on GPUs and TPUs.

Efficiency

Gradients as vectors and matrices unlock memory-optimized, hardware-accelerated training.

The Math Behind Vectorized Gradients

The Jacobian Matrix

Foundation of vectorized differentiation

$$f : \mathbb{R}^n \to \mathbb{R}^m$$

$$J[i,j] = \partial f_i \,/\, \partial x_j$$

Each row → **output component**
Each column → **input variable**

Captures how every output changes with respect to every input.

Chain Rule in Vector Form

Applied across layers of a neural network

Input → Linear Layer → Activation → Linear Layer → Output

$$d(g \circ f)/dx = (\partial g/\partial f) \cdot (\partial f/\partial x)$$

Jacobian product — not scalar multiplication

Instead of applying the chain rule manually for each parameter, vectorized differentiation uses matrix multiplication of Jacobians — allowing gradients to propagate efficiently through deep networks.

Backpropagation & Key Takeaway

Backpropagation

Applies the vectorized chain rule from output → input

1. **Forward Pass**
   Compute outputs for each layer
2. **Compute Loss**
   Measure prediction error
3. **Output Gradients**
   Calculate ∂Loss/∂output
4. **Propagate Backward**
   Apply Jacobian chain rule
5. **Update Parameters**
   Optimizer adjusts weights

Key Takeaway

Neural network training relies on vectorized gradient computation using Jacobian matrices and the chain rule. Backpropagation leverages these foundations to efficiently compute gradients across millions or billions of parameters — making modern deep learning possible.

VECTORIZED COMPUTATION ENABLES:

- Faster Training
- GPU Utilization
- Parallel Compute
- Memory Optimized
- Scalable to Billions
- Powers LLMs

#DeepLearning #MachineLearning #NeuralNetworks

AI & ML IN HEALTHCARE

Accelerating Drug Discovery, Diagnostics & Genomics

Protein Folding

AlphaFold 3

DeepMind's model predicts 3D protein structures from sequence with atomic accuracy — accelerating drug target identification from years to hours.

Diagnostics

Medical Imaging AI

CNN and Vision Transformer models (NVIDIA MONAI) detect tumors, diabetic retinopathy, and COVID-19 from CT/MRI scans with >95% AUC.

Genomics

Genomics & Precision Medicine

NVIDIA Clara Parabricks runs GPU-accelerated GATK genome variant calling in 18 min vs. 30 hrs on CPU — enabling real-time clinical genomics.

Drug Design

Drug Discovery (GNN)

Graph Neural Networks screen billions of molecular candidates. Schrödinger FEP+, Insilico Medicine, and Exscientia use HPC+ML pipelines.

Privacy AI

Federated Learning

Hospital systems train shared AI models without sharing patient data using NVFlare — preserving privacy while improving diagnostic models.

Simulation

Digital Twins

Heart & tumor digital twins (NVIDIA Omniverse + physics simulations) predict patient-specific treatment outcomes before procedures.

AI IN OIL & GAS AND GEOSCIENCE

Seismic Interpretation · Reservoir Modeling · Subsurface AI

OIL & GAS

Seismic Imaging (FWI)

Full Waveform Inversion on GPU clusters reduces seismic processing time 100x. ExxonMobil & Shell deploy NVIDIA A100 arrays for real-time imaging.

Reservoir Simulation

Physics-informed Neural Networks (PINNs) model subsurface fluid flow. CMG GEM and Eclipse simulators now integrate ML surrogate models.

Drilling Optimization

LSTM and reinforcement learning predict bit wear, mud weight, and pore pressure to prevent blowouts and reduce NPT.

LLM for Well Logs

Transformer models classify lithology and fluid type from wireline logs — replacing months of manual petrophysics work.

GEOSCIENCE

Climate Modeling (AI Emulators)

Google DeepMind's GraphCast forecasts 10-day global weather in <1 min. NVIDIA FourCastNet runs on supercomputers at 1 km resolution.

Earthquake Prediction

Deep learning on seismic waveforms detects microearthquakes and fault slip events. USGS and Caltech use EQTransformer models.

Satellite Image Analysis

CNN segmentation of SAR/multispectral imagery maps geological hazards, glacial retreat, and land subsidence in real time.

Subsurface Facies ML

3D CNN and GAN models generate synthetic geological facies models for uncertainty quantification in resource estimation.

AI-POWERED BIOMARKER DISCOVERY

Blood, CSF, and Fluid Biomarkers in AD and PD

Biomarkers are measurable biological indicators of disease state. AI dramatically accelerates biomarker discovery, validation, and integration into predictive models — enabling non-invasive or minimally invasive early diagnosis.

Alzheimer's Disease Biomarkers

CSF Biomarkers (Classical):

- Aβ42↓, p-Tau181↑, t-Tau↑ — validated AD signature
- ML classifiers on CSF triplet: AUC 0.93–0.97
- XGBoost on 200+ CSF proteins identifies novel AD markers

Blood (Plasma) Biomarkers:

- Plasma p-Tau217 — best single predictor (AUC 0.96)
- Plasma Aβ42/40 ratio predicted by ML from proteomics
- GFAP and NfL: neurodegeneration markers, ML-modeled
- AI builds composite blood panels rivaling PET accuracy

Parkinson's Disease Biomarkers

α-Synuclein (Seeding Amplification):

- Seed Amplification Assay (SAA) — 87% sensitivity for PD
- ML distinguishes PD from MSA and DLB using SAA kinetics
- Plasma NfL tracks motor and cognitive decline rate

Metabolomics & Proteomics:

- Urine and serum metabolomic profiling by ML
- Deep learning on mass spectrometry data identifies PD metabolic signature
- Dried blood spot analysis + CNN: low-cost screening tool
- Urine bis(monoacylglycerol)phosphate (BMP) — novel PD marker

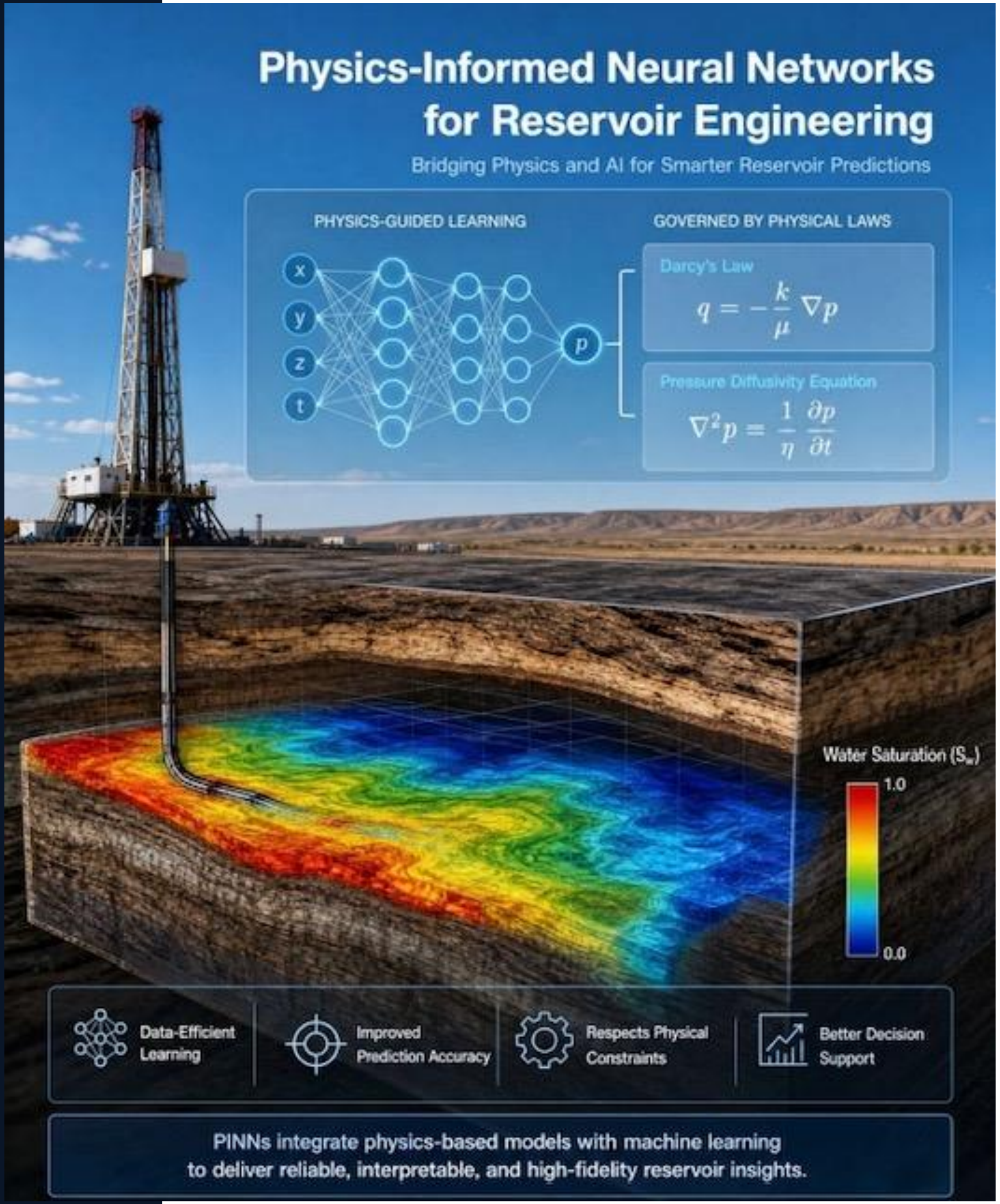
Physics-Informed Neural Networks

PINNs

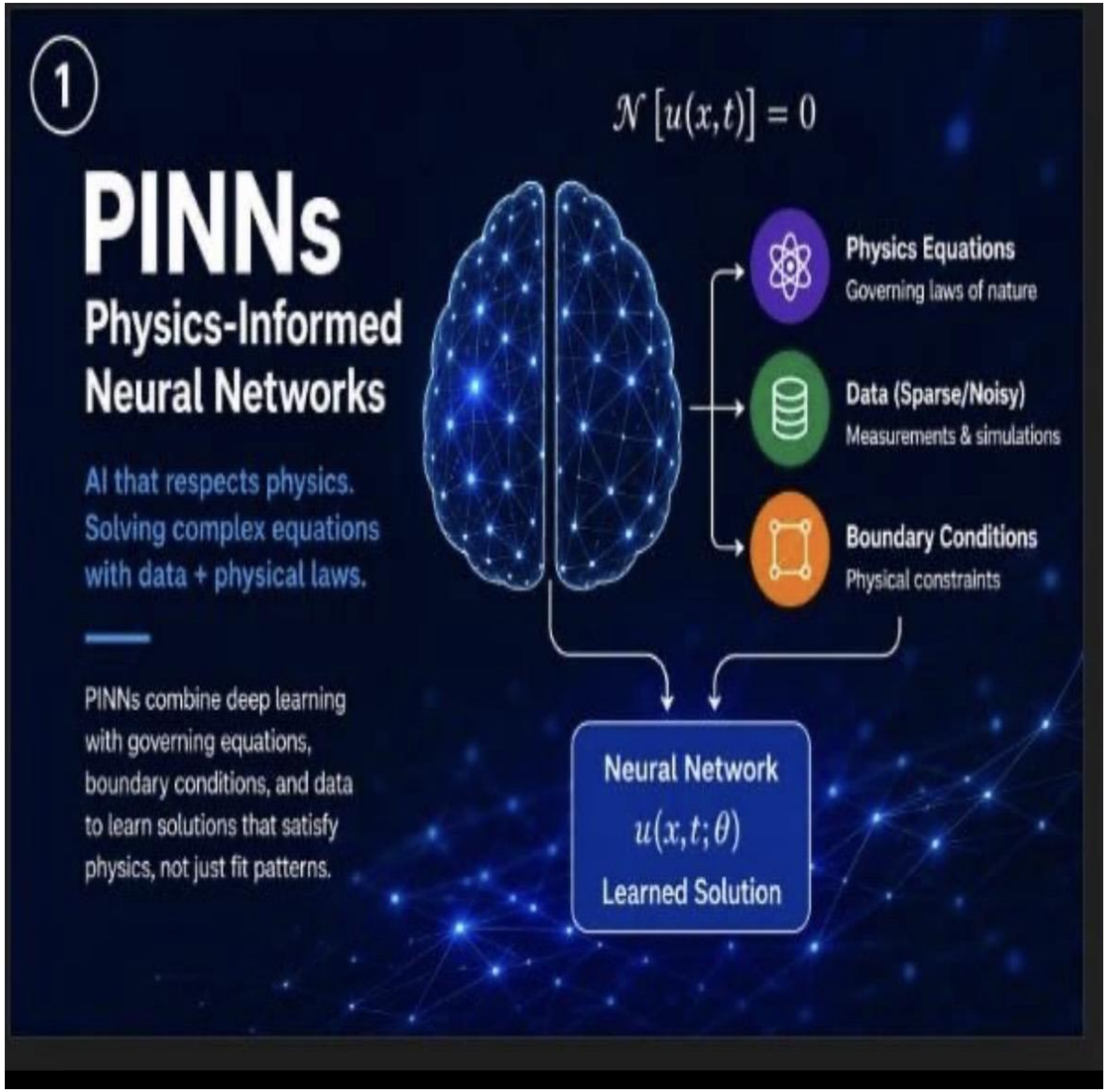
Where Physics Meets Artificial Intelligence

- Data-Driven + Physics-Guided Machine Learning
- Kernel Methods & Applications
- Oil & Gas | Medical Systems

AI-Assisted Nonlinear Applied Mathematics Research



Introduction to Physics-Informed Neural Networks



What is a PINN?

A Physics-Informed Neural Network is a deep learning model that encodes physical laws (PDEs, ODEs, conservation equations) directly into the training process — not just fitting data but satisfying the governing equations of the system.

Physics Equations

Navier-Stokes, Heat, Wave, Schrödinger...

Sparse/Noisy Data

Real measurements & sensor readings

Boundary Conditions

Physical constraints on domain edges

`Output: u(x,t;θ) — learned solution satisfying N[u]=0`

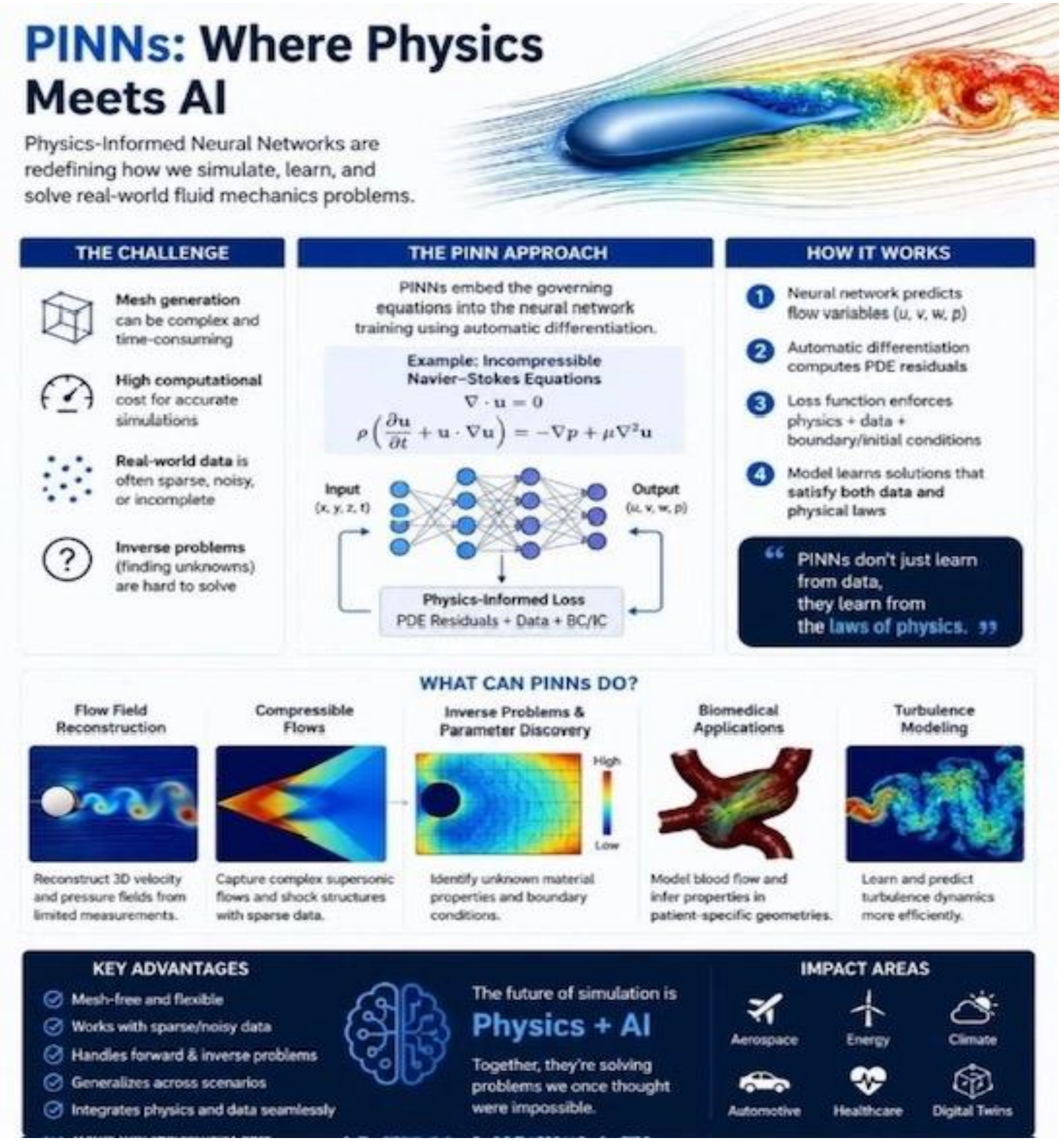
Why Data Alone Is Not Enough

Aspect	Pure Data-Driven ML	Physics-Guided (PINNs)
Data requirement	Massive labeled datasets	Works with sparse/noisy data
Generalization	Fails outside training dist.	Respects physical constraints
Interpretability	Black-box predictions	Governed by known equations
Conservation laws	Not guaranteed	Enforced by design
Extrapolation	Unreliable	Physics constrains the space
Inverse problems	Difficult	Native capability

Core Insight

Real-world scientific problems are governed by PDEs, conservation laws, and physical constraints. A model that ignores this physics may fit training data perfectly yet fail catastrophically in deployment, especially with sparse measurements or unseen scenarios.

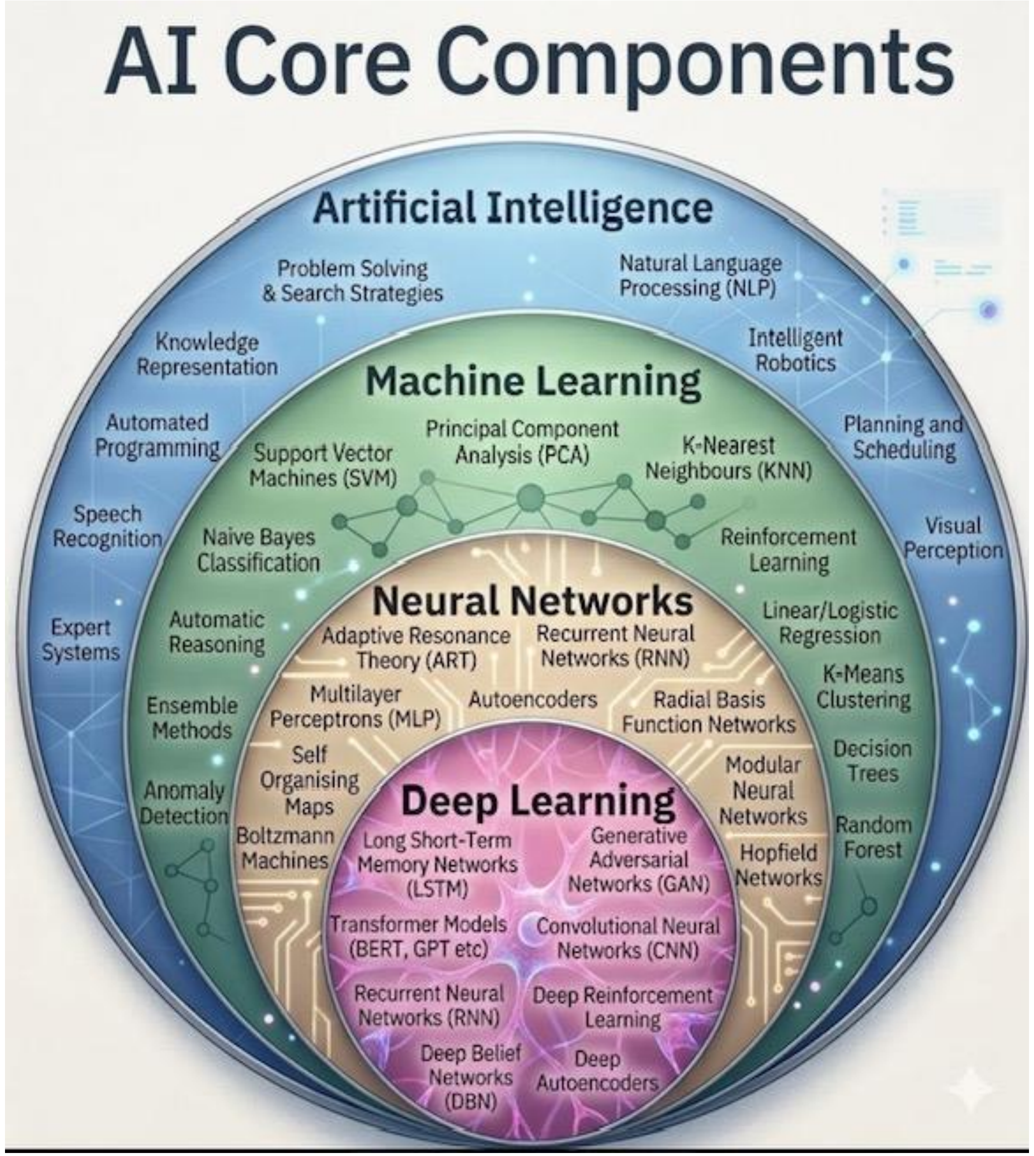
Physics-Guided Machine Learning



Data-Driven	Physics Only	Physics + ML (PINNs)
Learn patterns from observations	Analytical / numerical PDE solvers	Loss = L_data + L_physics + L_BC
Statistical correlations	FEM, FDM, CFD	Mesh-free & efficient
Requires large datasets	High computational cost	Works with sparse data
No physics guarantee	Requires known parameters	Solves forward & inverse

$L(\theta) = \lambda_1 \, L\_data + \lambda_2 \, L\_physics + \lambda_3 \, L\_boundary \rightarrow$ Minimize

Neural Network Foundations for PINNs



PINN Architecture

- Input Layer: (x, t) — spatial & temporal coords
- Hidden Layers: tanh/sin activations (smooth derivatives)
- Output Layer: $u(x,t)$ — solution field
- Auto-differentiation: $\partial u/\partial x$, $\partial^2 u/\partial x^2$ via backprop
- Physics residual computed at collocation points

Data Loss

$$L_{data} = \Sigma\ |u_{pred} - u_{obs}|^2$$

Match sensor measurements

Physics Loss

$$L_{phys} = \Sigma\ |N[u]|^2$$

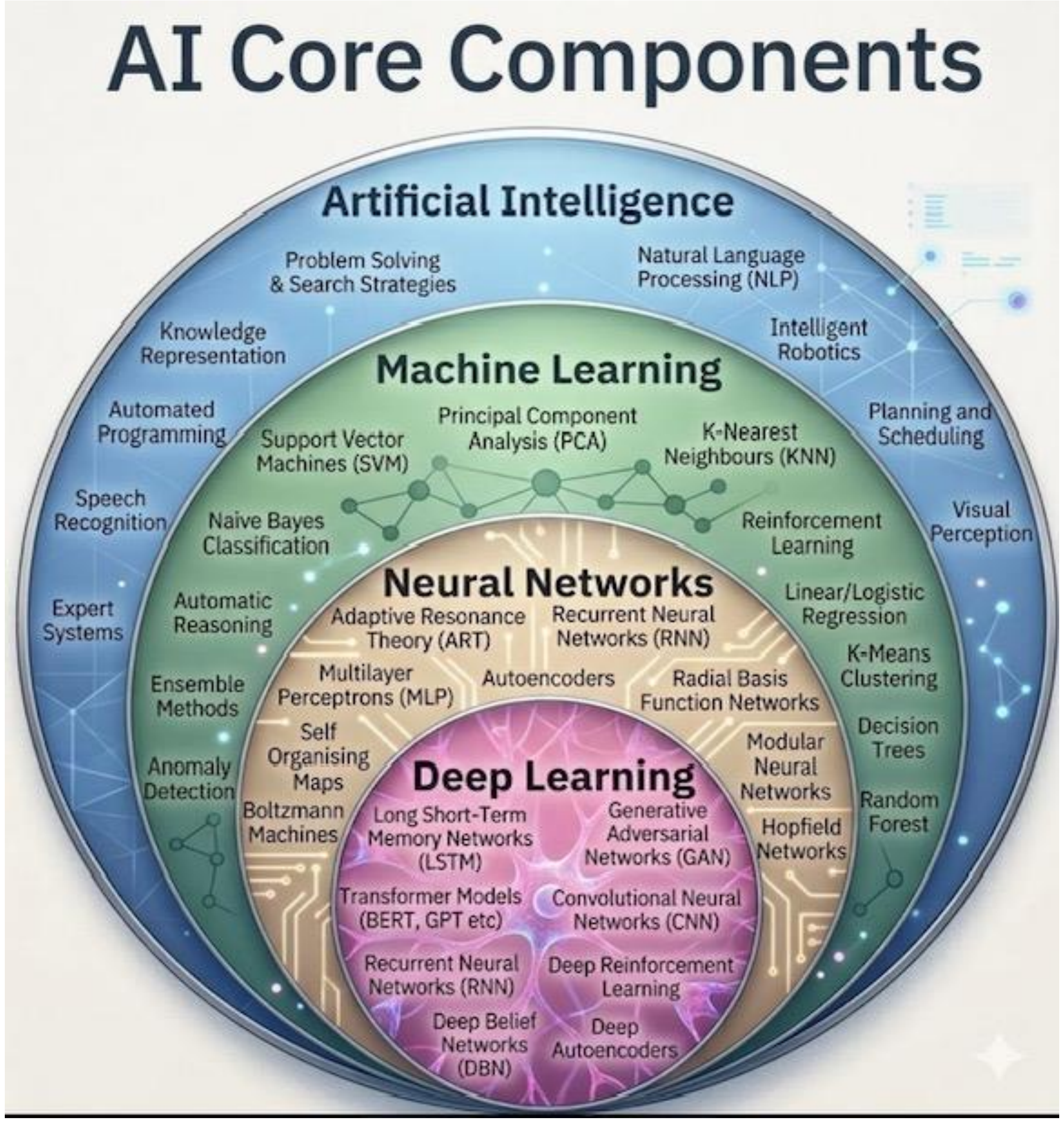
PDE residual at collocation pts

Boundary Loss

$$L_{bc} = \Sigma\ |u - u_{bc}|^2$$

Enforce boundary/initial cond.

AI Landscape: Where PINNs Fit



Deep Learning Core

PINNs leverage fully-connected deep networks with smooth activations (tanh, sin) for automatic differentiation of solution fields.

Supervised + Physics

Unlike standard supervised learning, PINNs are guided by physics residuals — a form of scientific inductive bias.

Radial Basis Networks

Kernel-based methods (RBF, GP) offer an alternative path to physics-informed learning with uncertainty quantification.

Hybrid Intelligence

PINNs bridge classical numerical methods (FEM/FDM) and modern ML, combining interpretability with flexibility.

Kernel Methods for Physics-Informed Machine Learning

Kernel methods provide a complementary framework to deep networks, enabling physics-informed learning with uncertainty quantification and guaranteed convergence properties.

Gaussian Process (GP)

$k(x,x') = \sigma^2 \exp(-\|x-x'\|^2/2\ell^2)$

- Probabilistic: provides uncertainty bounds
- Physics-informed GP: encode PDE as linear operator
- GP regression on PDEs: L*k(x,x') enforces physics
- Key for sparse data & sensor fusion

Radial Basis Functions (RBF)

$\phi(r) = e^{(-\varepsilon r^2)}$ or $(1+\varepsilon r^2)^{-\beta}$

- Meshless PDE collocation
- RBF-FD: local stencil approximation
- High accuracy for smooth solutions
- Natural for scattered data

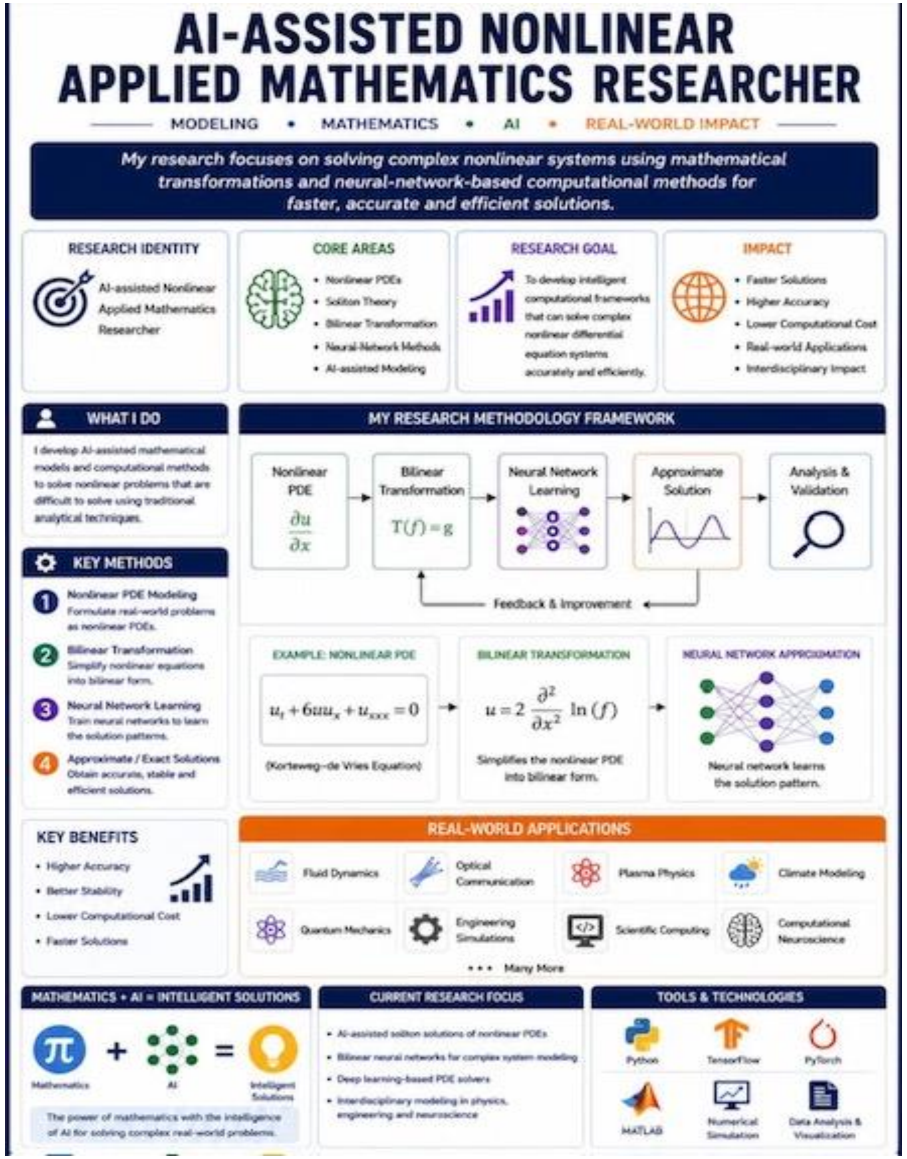
Deep Kernel Learning (DKL)

$k_\theta(x,x') = k_{base}(f_\theta(x), f_\theta(x'))$

- Neural network learns feature map
- GP on top provides calibration
- Physics constraints in kernel design
- State-of-art in physics-informed GPs

Kernel trick: map inputs to high-dimensional feature space → linear learning in infinite dimensions

Physics-Informed Kernels — Deeper Perspective



Operator-Valued Kernels

Define kernels over function spaces. If L is a PDE operator, design k_L such that samples from the GP automatically satisfy L[u]=0. Enables physics in the prior.

Kernel PDE Collocation

Place collocation points {x_j} in domain. Enforce residuals L[u](x_j)=0 as linear constraints in the GP posterior. Equivalent to RBF-based PDE solvers.

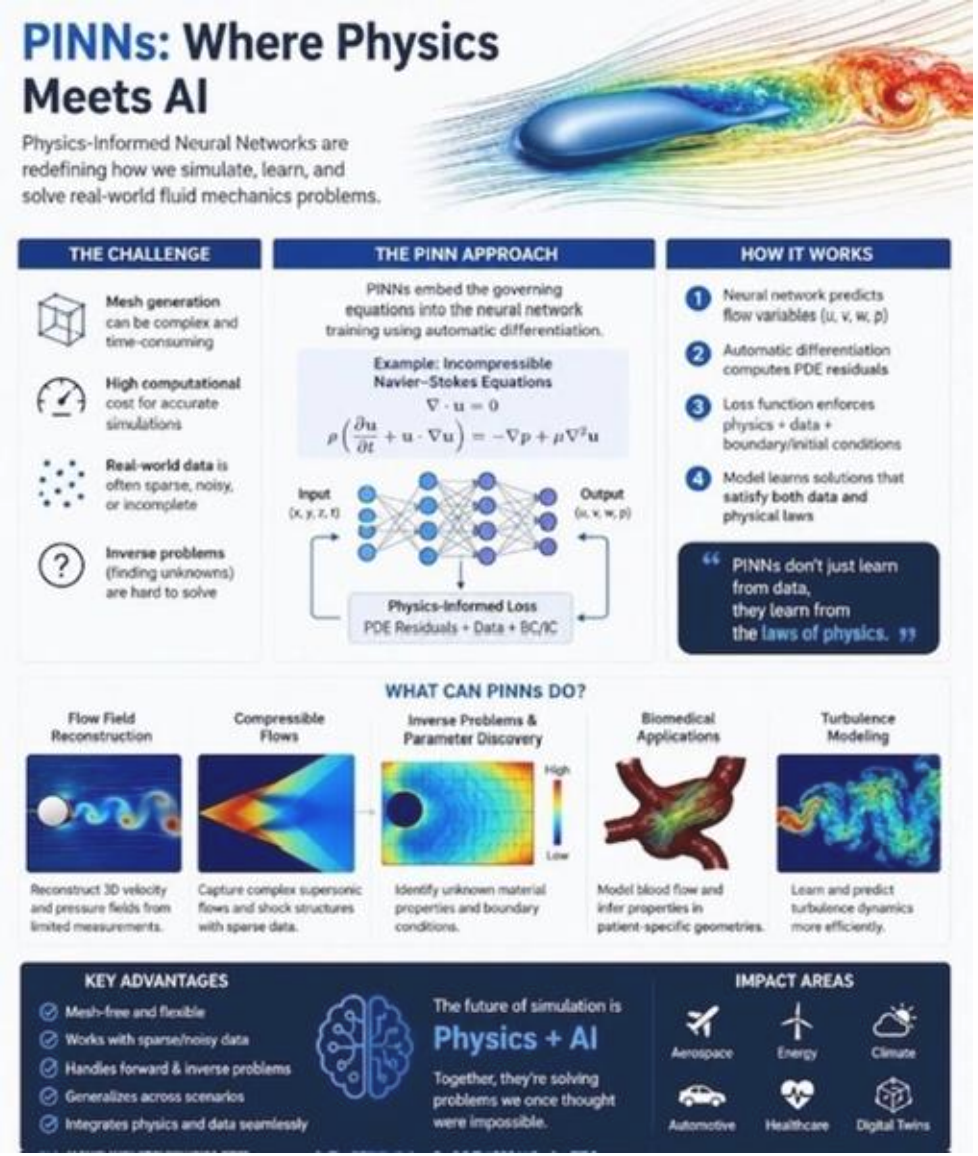
SVM + Physics Constraints

Support Vector Regression with PDE-derived features. Physics equations become additional support vectors, guiding the hyperplane in RKHS to respect governing laws.

Spectral Kernel Methods

Exploit Fourier/spectral decomposition of PDEs. Mercer kernels align with eigenfunction expansions — natural for periodic physics (wave equations, oscillators).

PINNs & Kernel Methods — Oil & Gas Applications



Reservoir Simulation

$$\partial(\phi\rho)/\partial t = \nabla\cdot(\rho k/\mu\ \nabla p) + q$$

PINNs solve the multiphase porous-media flow PDE with sparse well data. Kernel GPs quantify uncertainty in permeability fields. Reduce simulation cost by 60–80% vs FEM.

Well Production Forecasting

$$q(t) = \mathrm{PINN}(p_{wf},\ k,\ \phi,\ S_w)$$

Physics-informed RBF networks integrate Darcy flow with sensor time-series. Captures decline curves with minimal history data, enabling early field optimization.

Seismic Wave Inversion

$$\partial^2 u/\partial t^2 = c^2(x)\nabla^2 u \ \rightarrow\ \text{recover } c(x)$$

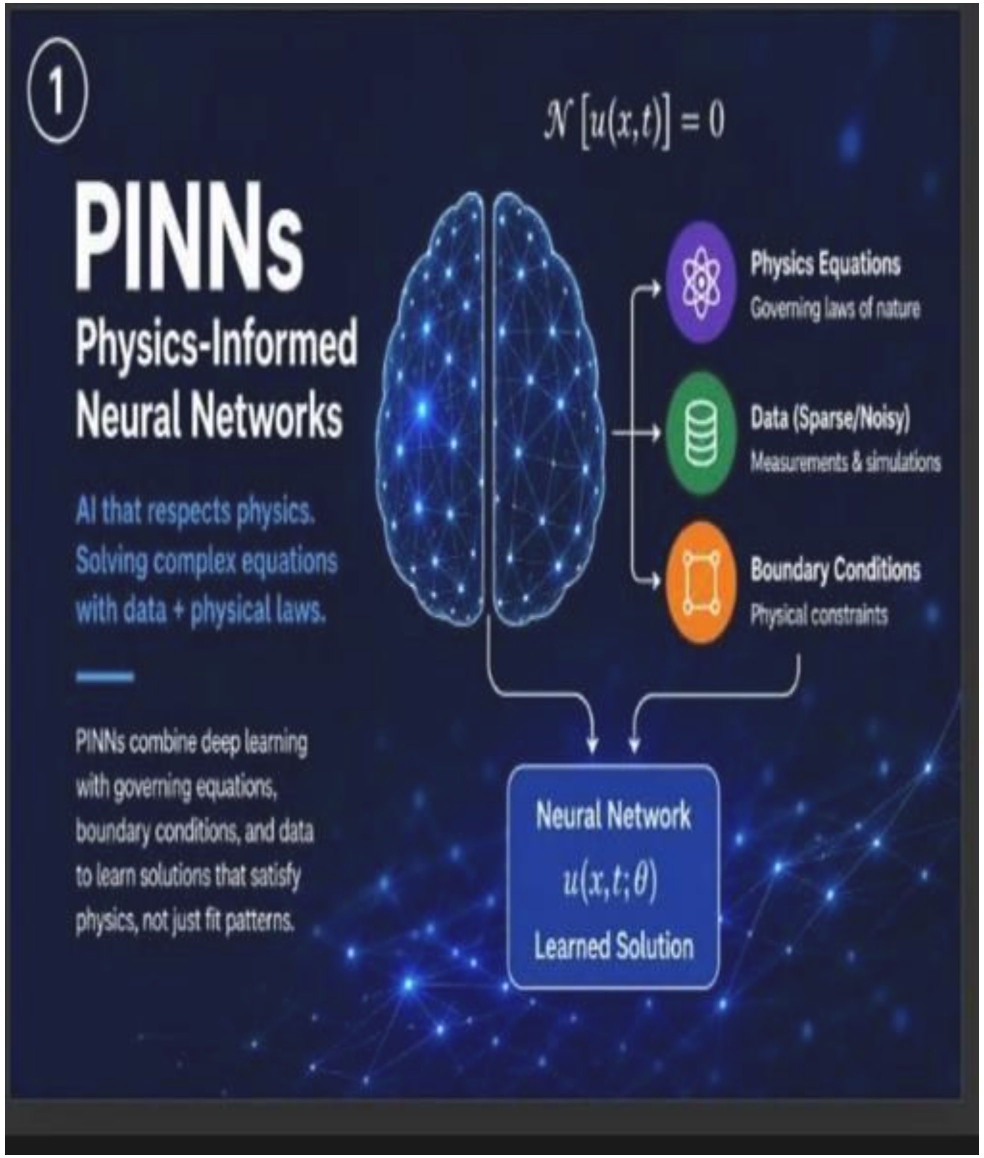
Solve inverse problem given sparse seismic recordings, infer subsurface velocity. Deep kernel learning provides uncertainty bounds essential for drilling risk assessment.

CO_2 Sequestration Monitoring

$$\partial C/\partial t + u\cdot\nabla C = D\nabla^2 C + R(C)$$

Monitor CO_2 plume migration with physics-informed GPs. Combines sparse injection well data with advection-diffusion-reaction PDE to forecast storage integrity.

PINNs & Kernel Methods — Medical Systems



Cardiovascular Hemodynamics

$\rho(\partial u/\partial t + u\cdot\nabla u) = -\nabla p + \mu\nabla^2 u$ (Navier-Stokes)

PINNs reconstruct 4D blood flow in arteries from sparse 4D-MRI data. Compute wall shear stress, pressure gradients, and aneurysm risk scores without invasive probes.

Tumor Growth Modeling

$\partial N/\partial t = \nabla\cdot(D\nabla N) + \rho N(1 - N/K) - \lambda N$

Physics-informed GP predicts patient-specific tumor evolution combining imaging data with reaction-diffusion equations. Guides radiation dosing and treatment timing.

Drug Delivery & Pharmacokinetics

$dC/dt = -(k_e + k_m)C + D_{input}(t)$

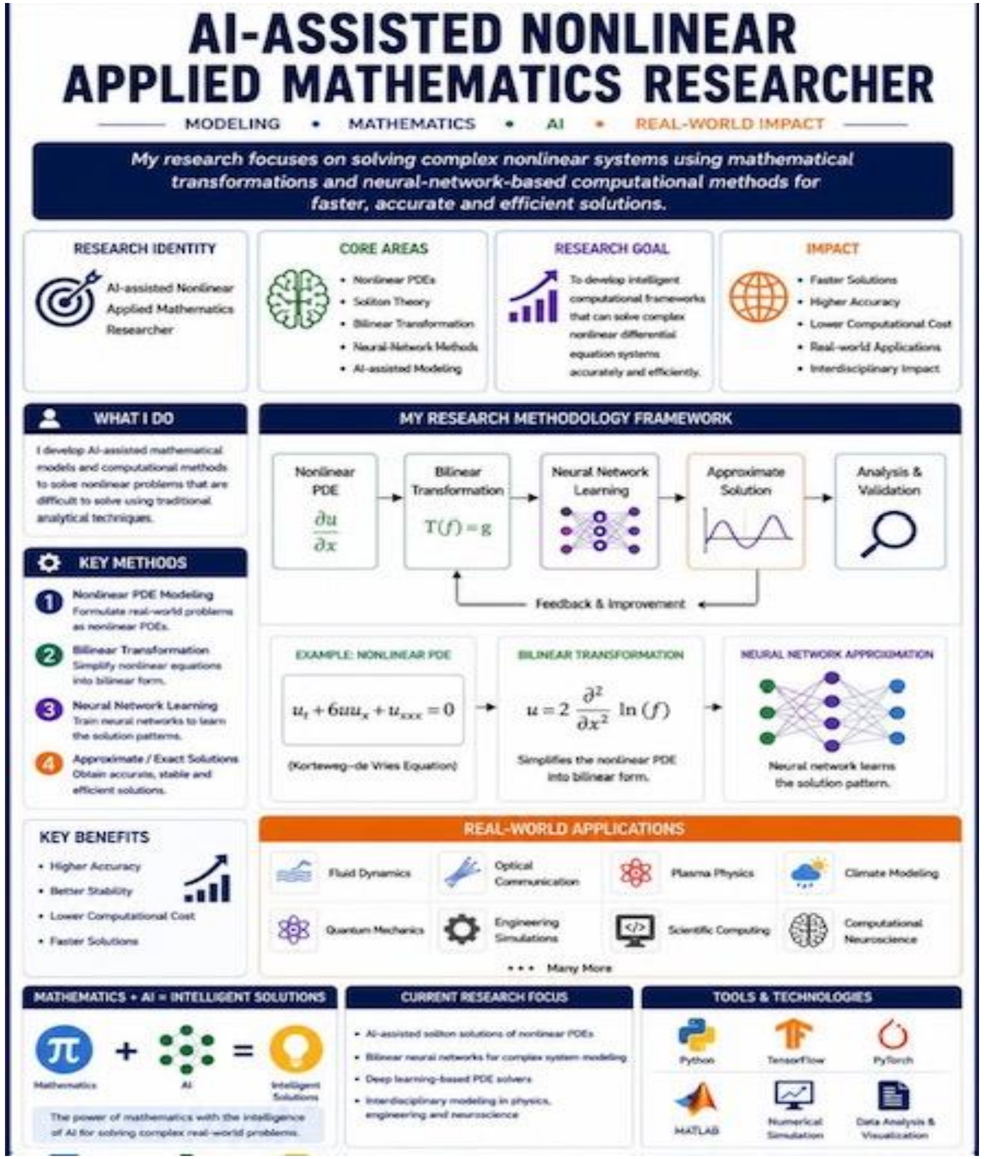
Kernel-enhanced PINNs model multi-compartment drug distribution using sparse blood sample data. Enables personalized dosing regimens with uncertainty quantification.

Brain Wave & EEG Modeling

$\partial^2 V/\partial t^2 = c^2\nabla^2 V + f(V)$ — neural field

Spectral kernel methods + PINNs for neural field equations. Reconstruct source localization from EEG electrodes. Applications in epilepsy monitoring and BCI.

Nonlinear PDEs: Bilinear Transformation + Neural Learning



1 Nonlinear PDE

$u_t + 6uu_x + u_{xxx} = 0$ (KdV equation)

Real-world physics: wave propagation, fluid dynamics, plasma physics, optical fibers.

2 Bilinear Transformation T(f)=g

$u = 2\partial^2/\partial x^2 \ln(f) \rightarrow$ linearizes KdV

Reduce complexity: bilinear forms simplify nonlinear equations into manageable structure for neural networks.

3 Neural Network Approximation

$\hat{u}(x,t;\theta) \approx u$ via $\min L(\theta)$

PINN learns the transformed solution pattern, inheriting physics from the transformation while leveraging data.

4 Analysis & Validation

$\|u_{PINN} - u_{exact}\| / \|u_{exact}\| < \varepsilon$

Relative L^2 error, convergence analysis, and comparison with finite difference / analytical solutions.

Conclusions & Path Forward

PINNs unify data and physics

Solve forward/inverse problems with sparse data by embedding governing equations in the loss function.

Beyond pure ML

Physics-guided learning generalizes better, conserves quantities, and is interpretable — critical for safety-critical applications.

Kernel methods enhance PINNs

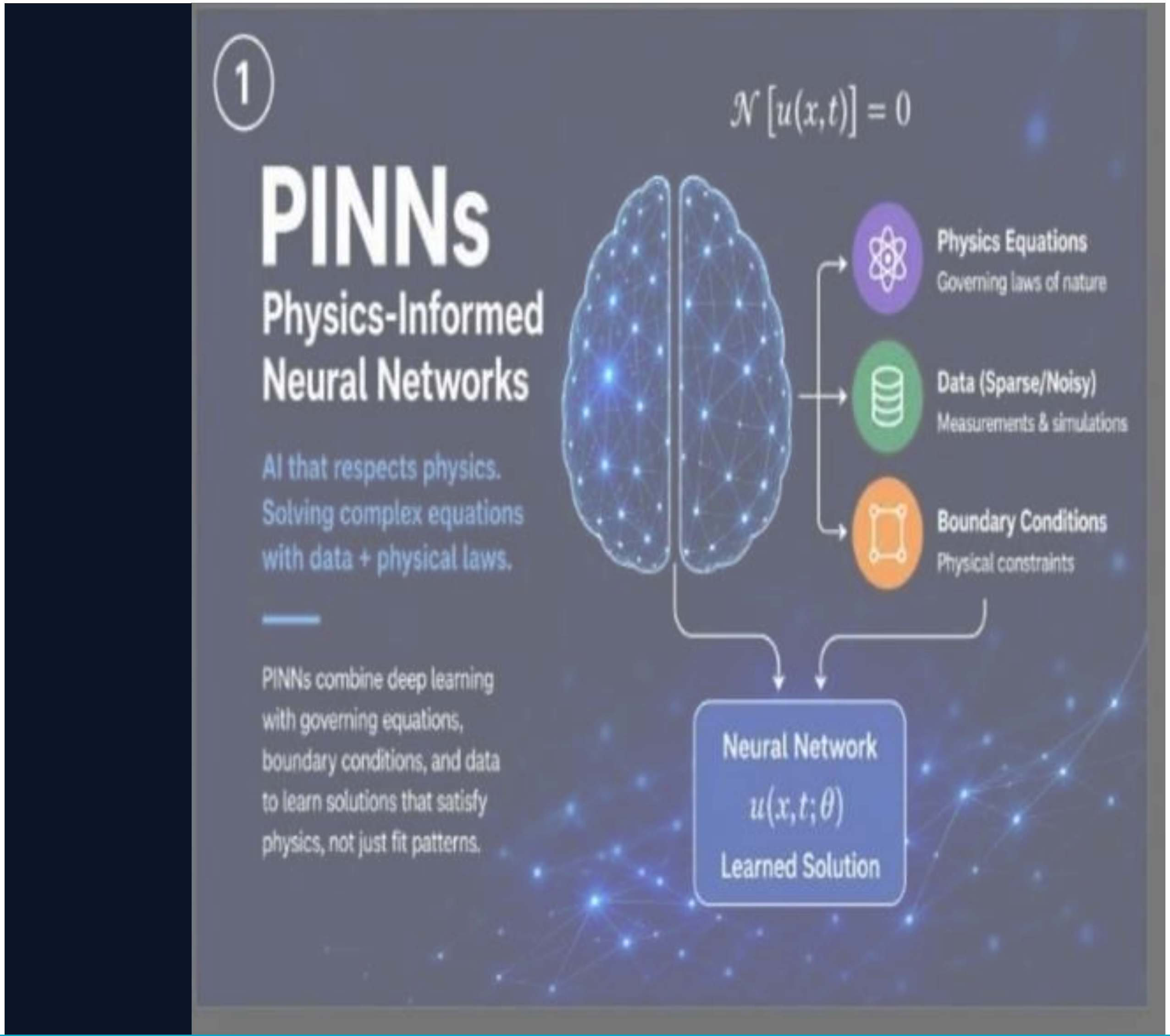
Gaussian processes, RBF, and deep kernels add uncertainty quantification and probabilistic inference to physics-informed models.

Oil & Gas impact

Reservoir simulation, seismic inversion, and production forecasting benefit from orders-of-magnitude speedups with physics constraints.

Medical breakthroughs

Patient-specific hemodynamics, tumor modeling, and drug delivery personalization with physics as the backbone of prediction.



Physics + AI = Intelligent Solutions to Problems We Once Thought Were Impossible

Automated Machine Learning (AutoML)

to Alzheimer's Disease Diagnosis and Prognosis

Pablo Guillén | Enrique Frias-Martinez

MIU City University | UNIR iTED

53,318

Unique Patients (NACC)

98.6%

Best Diagnosis Accuracy

92.8%

Best Prognosis Accuracy

Applied Artificial Intelligence · 2025 · Vol 39 · DOI: 10.1080/08839514.2025.2565166

Motivation & Problem Statement

The Challenge

- Alzheimer's affects 7M+ Americans & 8M+ EU citizens
- No effective treatments to slow progression
- Early detection is critical for intervention
- Traditional cognitive tests are biased & limited
- ML requires technical expertise — a barrier for clinicians

The AutoML Solution

- AutoML automates model selection & tuning
- Enables clinicians without deep ML expertise
- PyCaret: low-code, 25+ algorithms, integrated SHAP
- Explainability builds clinical trust
- Results comparable to or better than expert-built models

Alzheimer's Disease — Background & Stakes

7M+
Americans affected

8M+
EU citizens affected

NC → MCI → AD
Disease progression

0
Treatments slowing AD

Key Diagnostic Categories

NC — Normal Controls: no cognitive impairment

MCI — Mild Cognitive Impairment: early warning stage

AD — Alzheimer's Disease: full clinical diagnosis

Why ML Matters Here

- Analyzes clinical, cognitive & genetic data patterns
- Differentiates NC / MCI / AD with high accuracy
- Predicts disease progression years in advance
- XAI adds transparency critical in medical contexts

AutoML: PyCaret Framework

What is PyCaret?

- Open-source, low-code ML library in Python
- 25+ supervised learning algorithms
- 70+ automated open-source algorithms
- Integrated SHAP for model explainability
- Automated hyperparameter tuning & validation
- Ideal for tabular medical data (EHR, diagnostics)

Key limitation: PyCaret supports only Multilayer Perceptrons (no full DL)

Top Algorithms Used

XGBoost	Extreme Gradient Boosting
LightGBM	Fast, memory-efficient GBM
GBC	Gradient Boosting Classifier
RF	Random Forest Ensemble
ET	Extra Trees Classifier

NACC Dataset — Data Source & Features

53,318	195,196	1,024	2005–2025
Unique Participants	Total Instances	Features	Collection Period

Functional Activities Questionnaire (FAQ)

- BILLS — manage financial matters
- TAXES — handle complex finances
- PAYATTN — attention & concentration
- TRAVEL — independent travel ability

Clinical Dementia Rating (CDR)

- MEMORY — short & long-term memory
- ORIENT — time, place & person awareness
- JUDGMENT — decision-making ability
- COMMUN — community functioning

Data Preprocessing Pipeline

1

Missing Values

Features >50% missing removed. Median imputation for numerical; mode for categorical.

2

Feature Scaling

8 cognitive features standardized to mean=0, std=1 (MEMORY, ORIENT, JUDGMENT, COMMUN, BILLS, TAXES, PAYATTN, TRAVEL).

3

Encoding

Low-cardinality (<50 categories): one-hot encoding. High-cardinality: frequency encoding to minimize dimensionality.

4

SMOTE Oversampling

Applied within each training fold during cross-validation to address class imbalance (MCI underrepresented vs NC & AD).

Experiment Design — Two Complementary Studies

Experiment 1 — Diagnosis

Classify current cognitive state

- NC vs AD
- NC vs MCI
- MCI vs AD
- **NC vs MCI vs AD (multiclass)**

Features: 37–77 per scenario (n=152,629 total)

Experiment 2 — Prognosis

Predict cognitive state 4 years later

- NC vs AD
- NC vs MCI
- MCI vs AD
- **NC vs MCI vs AD (multiclass)**

Features: 52–54 per scenario (n=4,701 total)

Methodology Pipeline

NACC Data

Pre-processing

80/20 Train-Test Split

PyCaret Model Selection 10-fold CV + SMOTE

Evaluation & SHAP Explainability

Grid Search

Random grid search over hyperparameter space for each algorithm

Cross-Validation

10-fold stratified CV on training set with SMOTE per fold

Best Model

Retrained on full training data; evaluated on held-out 20% test set

Metrics

Accuracy, Precision, and Recall reported for all scenarios

Class Distribution & Imbalance Handling

Experiment 1 — Diagnosis

Scenario	NC	MCI	AD
NC vs AD	62.2% (94,933)	—	37.8% (57,590)
NC vs MCI	73.5% (94,933)	26.5% (34,106)	—
MCI vs AD	—	37.2% (34,106)	62.8% (57,590)
NC vs MCI vs AD	50.8% (94,933)	18.4% (34,196)	30.8% (57,590)

Experiment 2 — Prognosis (follow-up cohort)

Scenario	NC	MCI	AD
NC vs AD	45.6% (1,803)	—	54.4% (2,150)
NC vs MCI	70.7% (1,803)	29.3% (748)	—
MCI vs AD	—	25.8% (748)	74.2% (2,150)
NC vs MCI vs AD	36.2% (1,803)	15.0% (748)	48.8% (2,150)

SMOTE applied within each training fold to address MCI underrepresentation — synthetic samples never leak into validation sets

Explainability — SHAP (SHapley Additive exPlanations)

How SHAP Works

- Post-hoc, model-agnostic explainability
- Assigns a Shapley value to each feature
- Prediction = baseline + sum of SHAP values
- Positive SHAP → feature boosts prediction
- Negative SHAP → feature suppresses prediction
- Provides both local (per-sample) and global

```
y_i = y_base + f(x_i1) + f(x_i2) + … + f(x_in)
```

Key SHAP Findings

MEMORY	Most influential in ALL scenarios
COMMUN	High impact across NC/MCI/AD
JUDGMENT	Critical for disease staging
ORIENT	Key in MCI differentiation
TAXES/BILLS	Functional daily living indicators
Genetic Markers	Secondary role in prognosis

Guillén & Frias-Martinez | Applied Artificial Intelligence (2025) | DOI: 10.1080/08839514.2025.2565166

Results — Experiment 1: Diagnosis Performance

Best model accuracy by scenario (top 4 algorithms per scenario evaluated; bold = best)

NC vs AD

98.6%

Accuracy

Precision: 98.6%

Recall: 98.6%

Best: XGBoost / LightGBM / GBC

NC vs MCI

91.3%

Accuracy

Precision: 91.3%

Recall: 91.3%

Best: XGBoost / LightGBM

MCI vs AD

92.5%

Accuracy

Precision: 93.4%

Recall: 91.4%

Best: XGBoost

NC vs MCI vs AD

89.5%

Accuracy

Precision: 89.6%

Recall: 89.5%

Best: Extra Trees

🔑 ***All scenarios exceed 89% accuracy. NC vs AD achieves near-perfect classification at 98.6%.***

Results — Experiment 2: Prognosis Performance (4-Year Prediction)

Predict cognitive state 4 years after initial visit — best model accuracy per scenario

NC vs AD	NC vs MCI	MCI vs AD	NC vs MCI vs AD
92.8%	**82.7%**	**90.2%**	**81.4%**
Accuracy	Accuracy	Accuracy	Accuracy
Precision: 94.5%	Precision: 87.1%	Precision: 91.9%	Precision: 81.8%
Recall: 92.8%	Recall: 82.7%	Recall: 90.2%	Recall: 81.4%
Best: LightGBM	*Best: Extra Trees*	*Best: Extra Trees*	*Best: Extra Trees*

🔑 ***Extra Trees dominates prognosis; MCI-related scenarios are hardest due to smaller class representation.***

SHAP Feature Importance — Experiment 1 (Diagnosis)

Top features identified by SHAP across all four Experiment 1 scenarios (XGBoost & ET models)

Cognitive Assessments (Dominant Predictors)

- MEMORY — Top feature in ALL scenarios; high = NC, low = AD/MCI
- COMMUN — Community functioning; strong AD vs NC differentiator
- JUDGMENT — Problem solving; separates MCI from NC
- ORIENT — Time/place awareness; key in AD vs MCI

Daily Living Activities (Supporting Predictors)

- TAXES — Complex financial management; declines in AD
- BILLS — Basic finance; relevant in NC vs AD differentiation
- TRAVEL — Independent travel; functional independence marker
- PAYATTN — Attention; important in NC vs MCI models

Genetic & Visit Markers (Marginal Role)

- NGDSWGAC_ng00067 — Genome-wide study genetic marker
- PACKET_i / PACKET_t — Visit packet type indicators
- ADGCEXR_exome2 — Alzheimer genetics consortium marker
- Additional genetic features add marginal predictive value

SHAP Feature Importance — Experiment 2 (Prognosis)

SHAP insights from best models: LightGBM (NC vs AD) and Extra Trees (all other prognosis scenarios)

NC vs AD

Cognitive measures dominate. Genetic features (ADGCRND, NGDSWGAC) play limited role. MEMORY leads.

NC vs MCI

MEMORY remains top feature. Genetic markers ADGCRND_adc10 and NGDSEXAC_ng00079 show notable contributions — early detection signal.

MCI vs AD

Cognitive abilities dominate. Genetic influence minimal. COMMUN, JUDGMENT, MEMORY are top 3.

NC vs MCI vs AD

MEMORY, JUDGMENT, COMMUN top predictors. Genetic marker ADGCRND_adc10 enters 4th position.

Consistent finding: Cognitive testing (MEMORY, JUDGMENT, COMMUN) are primary predictors; genetic markers provide additional prognosis value.

Comparison with State of the Art (NACC Dataset)

All comparisons use the NACC dataset for consistency. AutoML vs manually-tuned expert models:

Study	Method	Scenario	Their Acc.	AutoML Acc.	Δ Gain
Alatrany et al. (2024)	SVM	NC vs AD	97.8%	98.6%	**+0.8%**
Alatrany et al. (2024)	SVM	NC vs MCI	88.6%	91.3%	**+2.7%**
Alatrany et al. (2024)	SVM	MCI vs AD	87.6%	92.5%	**+4.9%**
Alatrany et al. (2024)	SVM	NC vs MCI vs AD	85.2%	89.5%	**+4.3%**
Gupta & Kahali (2020)	SVM/RF	NC vs AD	92.0%	98.6%	**+6.6%**
Gupta & Kahali (2020)	SVM/RF	NC vs MCI vs AD	77.0%	89.5%	**+12.5%**
Mmadumbu et al. (2025)	LSTM-FNN	NC vs MCI vs AD	89.0%	89.5%	**+0.5%**
Tiwari et al. (2024)	Various	NC vs AD	84.0%	98.6%	**+14.6%**
Bucholc et al. (2023)	Hybrid ML	MCI vs AD (prog.)	87.5%	90.2%	**+2.7%**

AutoML performance improvement ranges from +0.5% to +14.6% over manually-tuned expert approaches on NACC.

Key Advantages of the AutoML Approach

① Systematic Algorithm Search

Automatically evaluates 25+ ML algorithms identifying robust alternatives that expert-driven approaches could overlook.

⑤ Competitive Performance

Matches or surpasses manually-tuned models, including DL baselines, using only tabular clinical data.

③ Standardized Evaluation

Consistent model benchmarking reduces risk of model-specific overfitting errors common in manual expert tuning.

④ Lower Technical Barrier

Clinicians without deep ML expertise can build, deploy, and interpret high-performance AD models in clinical settings.

② Integrated Explainability

Built-in SHAP integration detects feature interactions improving understanding of AD progression mechanisms.

⑥ Reproducibility & Standardization

Automated pipelines ensure consistent preprocessing, cross-validation, and reporting across research teams.

Limitations & Future Directions

Current Limitations

- PyCaret offers limited Deep Learning support (MLP only); no RNNs, LSTMs, or Transformers
- No subgroup error analysis by age, gender, or education (not natively supported in PyCaret)
- SMOTE cannot fully replicate complexity of original minority-class data distributions
- Class imbalance (small MCI group) remains challenging for NC vs MCI and MCI vs AD scenarios
- SHAP indicates statistical contribution — clinical interpretability requires domain expert validation

Future Directions

- Integrate DL techniques (LSTMs, Transformers) into AutoML pipelines for longitudinal data
- Develop subgroup error analysis capabilities for age, gender, and education stratification
- Create clinical decision-support visualizations that communicate SHAP to non-technical clinicians
- Extend AutoML + XAI to other neurodegenerative diseases (Parkinson's, ALS, Huntington's)
- Incorporate multimodal data (MRI, genetic, clinical) into future AutoML AD pipelines

Conclusions

01 AutoML (PyCaret) delivers competitive AD diagnosis and prognosis, reaching 98.6% accuracy on NC vs AD — matching or exceeding expert-built models.

02 SHAP consistently identifies MEMORY, COMMUN, and JUDGMENT as primary predictors across all scenarios, aligned with clinical knowledge.

03 AutoML improves model performance by 0.8% to 12% over manually-tuned approaches, while dramatically lowering the technical knowledge barrier.

04 Genetic markers contribute marginally in diagnosis but show emerging relevance in prognosis, particularly for NC vs MCI early detection.

05 The AutoML + XAI framework provides a deployable, explainable path for clinical neurodegenerative disease decision-support systems.

KlinicSync

Genómica de Precisión & Inteligencia Artificial en Oncología

Innovación molecular para México y América Latina · 2026

www.klinicsync.com | contacto@klinicsync.com | Mérida, Yucatán, México

KlinicSync

Quiénes Somos

2020 Fundación

9+ Años de experiencia

8+ Laboratorios aliados

500+ Oncólogos atendidos

Empresa Mexicana Especializada en Oncología de Precisión

- Promoción, comercialización, distribución y logística de pruebas oncológicas de diagnóstico molecular.
- Acreditados para promoción de laboratorios de perfilamiento genómico en América y Europa.
- Conectamos a pacientes y oncólogos mexicanos con las pruebas genómicas más innovadoras del mundo.
- Facilitamos el acceso a medicina de precisión con tratamientos personalizados para pacientes con cáncer.
- Socios con Tempus AI, Foundation Medicine, Caris Life Sciences, INVITAE, Burning Rock Dx, Pangea Oncology, Digistain y Wren Laboratories.

Nuestros Servicios

Perfilamiento genómico integral para oncólogos y pacientes en México y América Latina

Perfilamiento Genómico

Secuenciación NGS de tumores sólidos y líquidos. Paneles de 100 a 648+ genes.

Biopsia de Tejido & Líquida

Pruebas en FFPE, sangre periférica y otros fluidos para diagnóstico completo.

Biomarcadores Moleculares

MSI, TMB, HER2, PD-L1, BRCA, fusiones génicas y variantes de empalme.

Riesgo de Recurrencia

Prosigna, Digistain: predicción a 5-10 años en cáncer de mama.

Soporte Clínico

Enlace científico entre laboratorio y oncólogo. Gestión de pedidos y resultados.

Acceso a Mercado

Acceso privado a laboratorios líderes globales desde México.

Laboratorios Aliados de Clase Mundial

TEMPUS AI

xT (648 genes), xE (Exoma), xR (Transcriptoma)
xF+ (523 genes líquida), MRD, HER2, PDL1, HRD

Foundation Medicine

FoundationOne CDx (324 genes)
FoundationOne Liquid CDx · HEME (406 genes)

INVITAE

Panel Multicáncer Hereditario Germinal
70 genes de predisposición hereditaria

Burning Rock Dx

OncoScreen Plus (520 genes NGS)
CanCatch MRD · OncoCompass (168-520 genes)

Pangea Oncology

Estudio Combinado Tumores Sólidos
Biopsia Líquida · Carcinoma Urotelial

Prosigna / Digistain

PAM50 Breast Cancer Assay · nCounter
Espectrometría IR - Recurrencia 5-10 años

Wren Laboratories

NETest (Tumores Neuroendocrinos)
PROSTest · NETest + PPQ - qPCR

MedGenome

Perfilamiento genómico para poblaciones latinoamericanas

IA & Machine Learning en Oncología

Investigación aplicada para México y América Latina

Diagnóstico Asistido por IA

Modelos de ML para clasificación tumoral, predicción de respuesta terapéutica y estratificación de riesgo en oncología.

Procesamiento de Datos Genómicos

Pipelines de análisis automatizados de variantes, fusiones génicas y expresión génica a gran escala con Deep Learning.

Bases de Datos Latinoamericanas

Desarrollo de repositorios genómicos representativos de la diversidad genética de México y Latinoamérica para modelos más precisos.

Investigación en IA para Latinoamérica

KlinicSync lidera la aplicación de inteligencia artificial y aprendizaje automático en oncología de precisión

Modelos Predictivos de Respuesta Terapéutica

Algoritmos de ML entrenados con datos multi-ómicos para predecir respuesta a inmunoterapia, quimioterapia y terapias dirigidas en poblaciones latinas.

Clasificación Tumoral Automática

Redes neuronales para identificar subtipos moleculares tumorales a partir de datos de expresión génica (RNA-seq) y perfiles de variantes somáticas.

Análisis de Variantes de Significado Incierto (VUS)

Modelos de interpretación patogénica de variantes usando bases de datos clínicas, estructurales y evolutivas con NLP y deep learning.

Detección Temprana de Cáncer por ctDNA

IA aplicada a biopsia líquida para detectar señales tumorales a muy baja frecuencia alélica, habilitando diagnóstico y monitoreo no invasivo.

Juntos Avanzamos hacia la Medicina de Precisión

en México y América Latina

🌐 www.klinicsync.com

✉ contacto@klinicsync.com

📍 Mérida, Yucatán, México · RFC: KLI200622TM4

Thank You

Questions & Discussion