Roma Tre

Tutorial 1: Machine learning bases and advanced applications for nanoindentation data analysis

Edoardo Rossi
Roma Tre University, Department of Civil, Computer Science, and Aeronautical Technologies engineering, Rome, Italy

1

Run it yourself: the repository

MecaNano

Open-source github.com/edrossi93/mecanano-ml-nanomechanics.

One click Binder / Colab: no install.

Follow along open notebook 00 and run cells as we go.

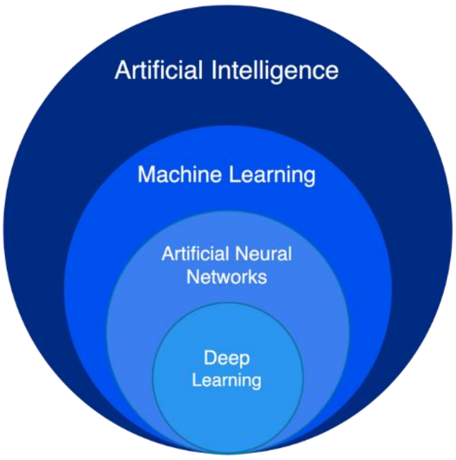
repo

Binder · nb 00

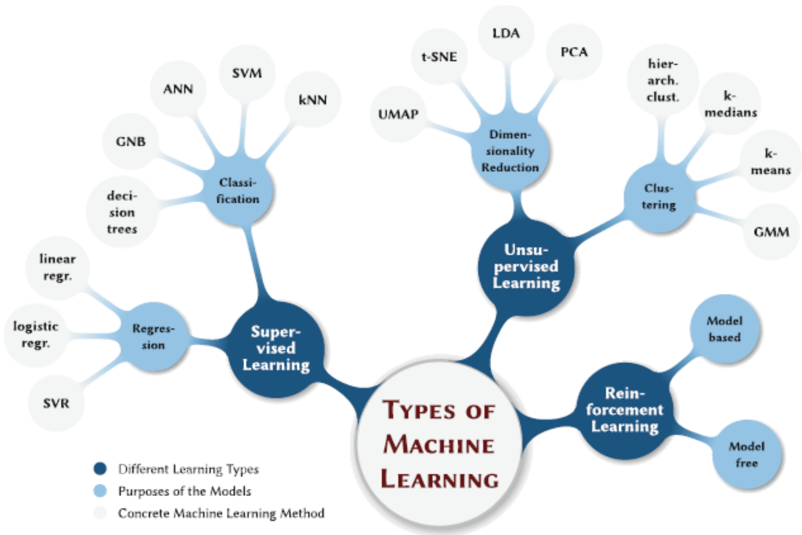
2

What is machine learning?

- **Artificial Intelligence (AI)** is the broad field that aims to create machines capable of intelligent behavior.
- **Machine Learning (ML)** is a subset of AI that focuses on algorithms that learn from data.
- **Artificial Neural Networks (ANNs)** are models inspired by the human brain, forming the foundation of many ML systems.
- **Deep Learning (DL)** refers to neural networks with many layers, capable of learning complex patterns in large datasets.



Types of Machine Learning



S. Sandfeld, The "Materials Data Science" book (2024, Springer)

The Machine Learning Workflow

MecaNano COST Action CA21121

1. Data collection
Data must be curated and validated from multiple sources.

2. Pre-processing
Outlayer removal, data scaling, dimensionality reduction (PCA)

3. Modelling
Choice of the appropriate ML algorithm.

4. Learning
Training of the ML model
(weights adjustments in NN or hyperparameters)

6. "Error Analysis"
Testing the ML model with new test data

7. Results

S. Sandfeld, The "Materials Data Science" book (2024, Springer)

Machine Learning for Nanoindentation - Edoardo Rossi - MecaNano WG4 – Tutorial 1 5

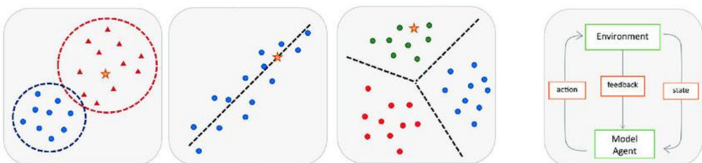
5

Supervised vs. Unsupervised Learning

MecaNano COST Action CA21121

- **Supervised Learning**
  - Learns a function from **labeled data**:
    $\{(x_1, y_1), \ldots, (x_n, y_n)\}$
  - Goal: minimize prediction error
    $$\min \theta \sum_{\{i=1\}}^{n} L(f(x_i;\ \theta), y_i)$$
  - Tasks:
    - Classification (e.g., phase = WC or binder)
    - Regression (e.g., hardness prediction)
- **Unsupervised Learning**
  - Learns from **unlabeled data**:
    $\{x_1, x_2, \ldots, x_n\}$
  - Goal: find hidden structure (clusters, manifolds)
  - Methods:
    - Clustering (e.g., K-means, GMM)
    - Dimensionality reduction (e.g., PCA, t-SNE)

Machine Learning
- Supervised Learning
  - Classification
  - Regression
- Unsupervised Learning
  - Clustering
- Reinforcement Learning



Machine Learning for Nanoindentation - Edoardo Rossi - MecaNano WG4 – Tutorial 1 6

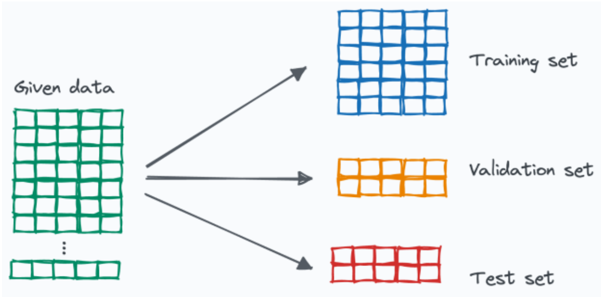
6

Training, Validation, and Testing: Managing Your Dataset

- Training set: Used to learn the model parameters
- Validation set: Used to tune hyperparameters and prevent overfitting
- Test set: Used only at the end to evaluate generalization

- Training ≈ 60%−70%
- Validation ≈ 15%−20%
- Test ≈ 15%−20%

Cross-validation (e.g. K-fold): Rotate training/validation sets across K partitions



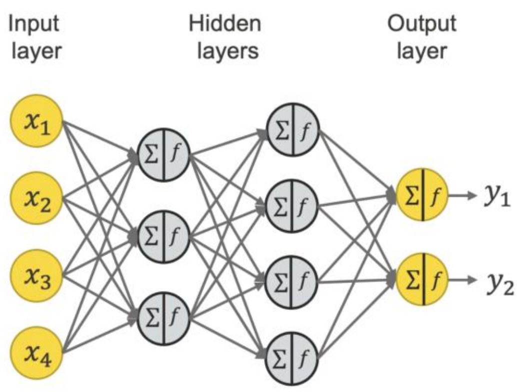
Deep Learning: Overview

- Deep learning is a subfield of machine learning based on **artificial neural networks (ANNs)** with many layers. It enables the **automatic extraction of features** from raw, high-dimensional data (e.g., images, signals, curves).
- A deep model learns a **composition of functions**:

$$f(x) = f_L\left(f_{\{L-1\}}(\dots f_{1(x)})\right)$$

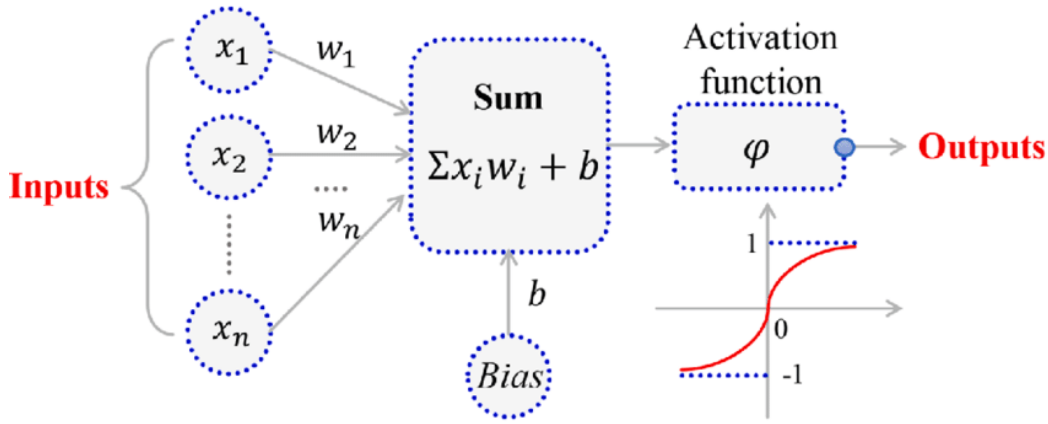
- where each layer f_l is a non-linear transformation with trainable parameters.

Artificial Neural Network (ANN)



The Artificial Neuron: From Inputs to Activated Output

MecaNano

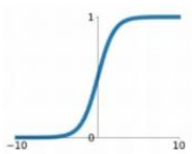


9

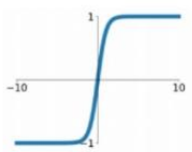
Activation functions

MecaNano

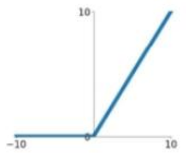
Sigmoid
$\sigma(x) = \frac{1}{1+e^{-x}}$



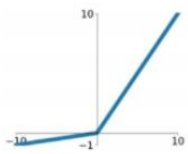
tanh
$\tanh(x)$



ReLU
$\max(0, x)$

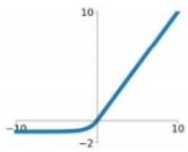


Leaky ReLU
$\max(0.1x, x)$



Maxout
$\max(w_1^T x + b_1, w_2^T x + b_2)$

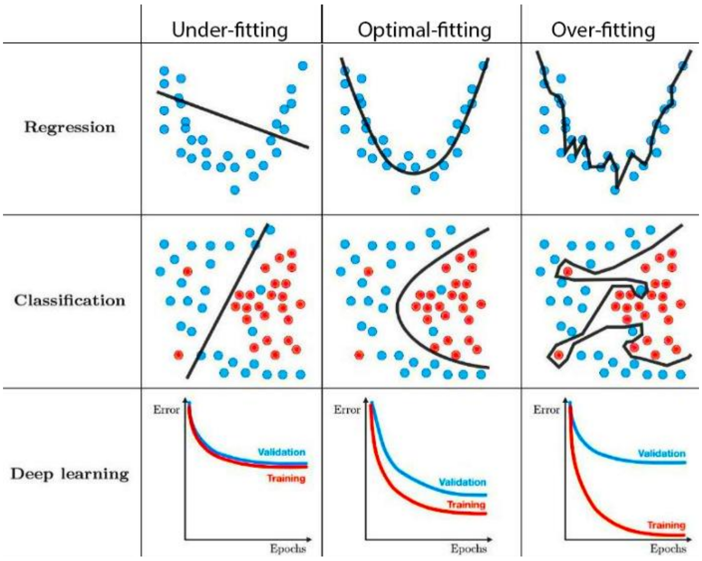
ELU
$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$



10

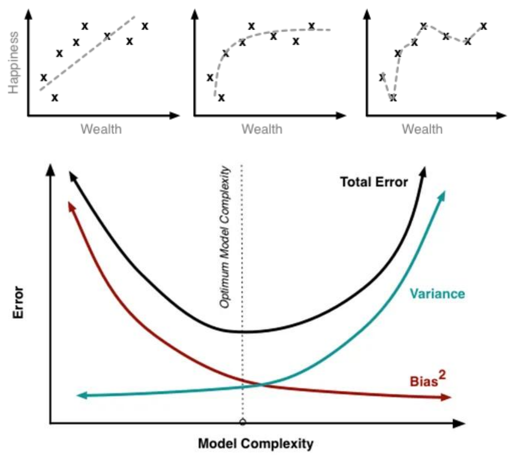
Underfitting / Overfitting

- **Underfitting**: Model too simple, fails to capture the structure of the data
- **Optimal fit**: Model generalizes well, captures patterns without memorizing noise
- **Overfitting**: Model is too complex, fits the training data perfectly but performs poorly on unseen data
- Training error:
  $$L_{train} = \frac{1}{n}\sum_i (f(x_i) - y_i)^2$$
- Generalization gap:
  $$Gap = L_{val} - L_{train}$$



Generalization: The Core Goal

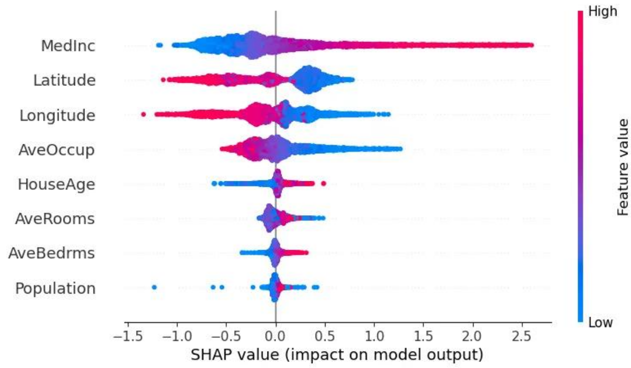
- In supervised learning, we don't care about performance on the training set.
  We care about **how well the model performs on unseen data.**
- This is the **generalization error**:
  $$\varepsilon_{gen} = \mathbb{E}_{x,y\sim\mathcal{D}_{test}}[L(f(x), y)]$$
- A model that memorizes data (overfits) is not generalizing.
  A model that underfits never captures the signal.



Interpreting Model Predictions Feature by Feature

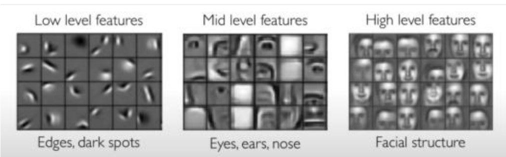
- SHAP (SHapley Additive exPlanations) assigns each feature an importance value for a specific prediction. It explains how much each feature **contributes to the prediction's accuracy** relative to a baseline.

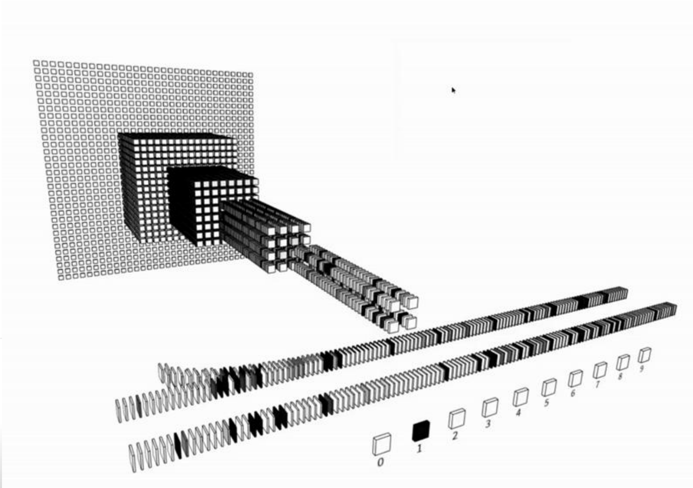
SHAP Summary Plot (global explanation)
Beeswarm plot:
 Y-axis: features
 X-axis: SHAP values
 Color: feature value (low to high)
 Shows which features matter globally and how they affect predictions



CNN (Convolutional Neural Networks)

- A CNN is a neural network architecture designed to process **grid-structured data** (e.g., images, 2D maps).
- It learns **spatial hierarchies of features** through layers of convolutions, pooling, and nonlinear activations.
- No need for manual feature engineering (useful for complexity)





Lee et al., ICML 2009 (DOI: 10.1145/1553374.1553453), adapted from A. Amini

Main ML libraries in Python

1. Scikit-Learn

A simple and efficient tool for data mining and data analysis, built on NumPy, SciPy, and matplotlib.

- **Features**:
  1. Easy-to-use interfaces for various machine learning algorithms.
  2. Preprocessing tools and utilities.
  3. Comprehensive documentation and active community support.

2. TensorFlow

Open-source library developed by Google for deep learning and neural networks.

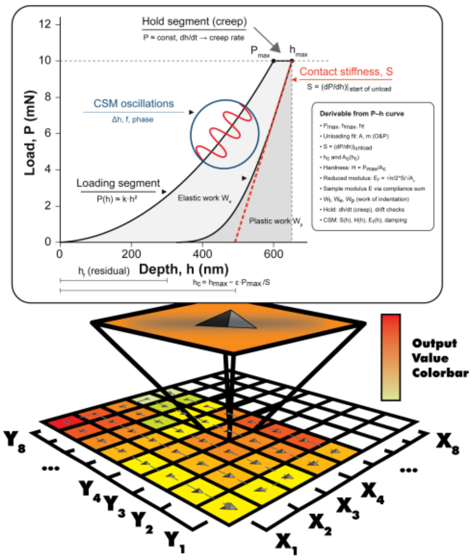
Features:
1. Flexibility to build and train custom machine learning models.
2. Support for both CPU and GPU computing.
3. High-level APIs like Keras for easier model building.

3. PyTorch

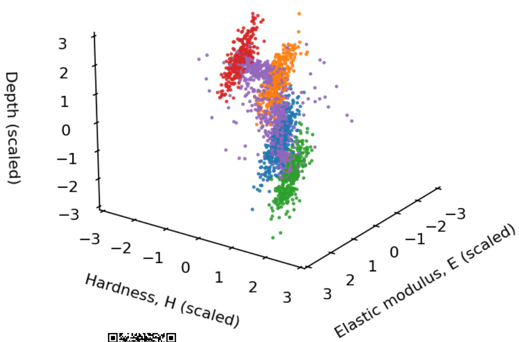
An open-source deep learning framework developed by Facebook's AI Research lab, used for applications such as computer vision and natural language processing.

- **Features**:
  1. Dynamic computation graphs for flexibility and speed.
  2. Strong GPU acceleration support.
  3. Integration with Python for easy and intuitive coding.

Seeing structure in high-dimensional data



Dimensionality reduction
(PCA, t-SNE, UMAP)

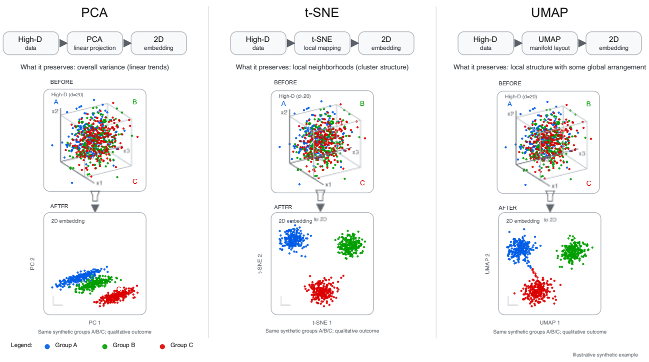


▶ Run it: notebook 08

scan or click · Colab, no install

Seeing structure in high-dimensional data

MecaNano COST Action CA21121



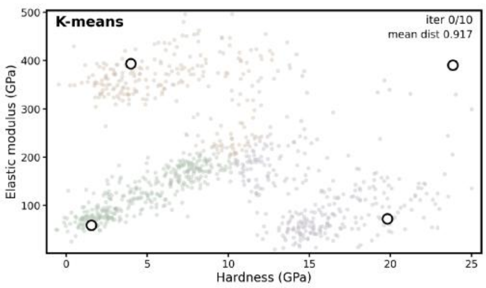
- **PCA**: a linear rotation that keeps the directions of greatest variance; fast, but sees only linear structure
- **t-SNE**: nonlinear; places similar curves together by minimising the **KL divergence** (how far the 2-D neighbour-probabilities sit from the high-D ones). Clusters are meaningful; the gaps between them are not
- **UMAP**: nonlinear like t-SNE but faster, and keeps more of the global layout

The two pillars of unsupervised clustering

MecaNano COST Action CA21121

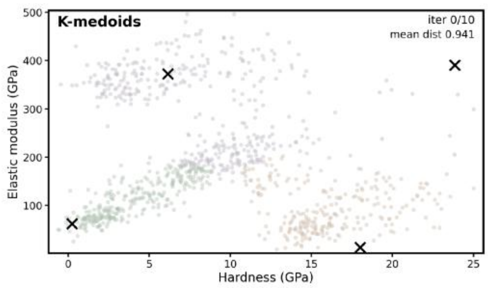
k-means partitions the indents into K groups, each summarised by a centroid (the mean H, E ... of its members), and iterates until the within-cluster scatter $\Sigma\|x_i-\mu_j\|^2$ stops shrinking. Fast and scalable to 10^5 indents, but it assumes round, equal-size clusters and depends on the starting seeds.

$$Min \sum_{j=1} \sum_{x_i \in C_j} \| x_i - \mu_j \|^2$$



k-medoids does the same, but each cluster centre is a real measured indent (a medoid), not an average. That makes it robust to outliers and usable with any distance metric: at higher computational cost.

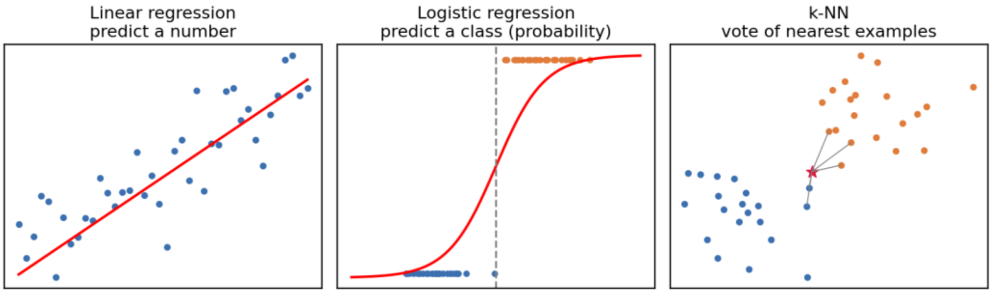
$$Min \sum_{j=1} \sum_{x_i \in C_j} d(x_i, m_j)$$



Use k-medoids when the cluster centre must be a physically-measurable point: a representative nanoindentation test you can re-examine: or when robustness to outliers and non-Euclidean distances matters.

Supervised **basics**: regression, logistic, k-NN

MecaNano



- **Linear regression**: fit a straight line y = w·x + b by least squares (minimise the summed squared errors) to predict a continuous value, e.g. hardness from modulus
- **Logistic regression**: squash the linear score through a sigmoid σ(z)=1/(1+e^(−z)) into a probability P∈[0,1]; predict the class when P > 0.5
- **k-NN**: no training; label a new indent by majority vote of its k nearest neighbours (small k overfits noise, large k oversmooths)

Clustering depth: **GMM** and choosing k

MecaNano

GMM: soft clustering

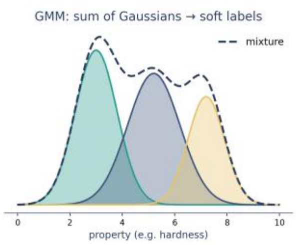
- Models the data as a **mixture of Gaussians**, $p(x)=\Sigma\, \pi_k N(x|\mu_k,\Sigma_k)$: a weighted sum of bell curves, one per phase
- Gives each indent a **posterior probability** of belonging to every phase, not a hard label

Reading uncertainty

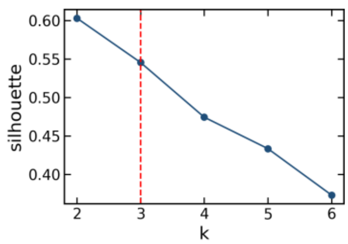
- The posterior splits near **phase boundaries** where responses overlap: it shows where the model is unsure

Choosing k

- **Silhouette** = (b−a)/max(a,b): how much tighter a point sits in its own cluster (a) than the nearest other (b); its peak sets k (here k = 3)



GMM



silhouette vs k

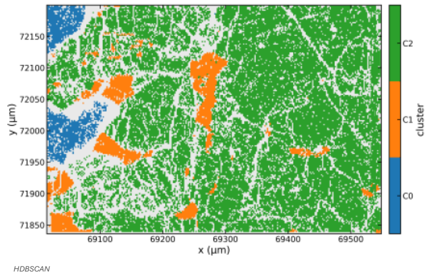
Density clustering & bagging vs boosting

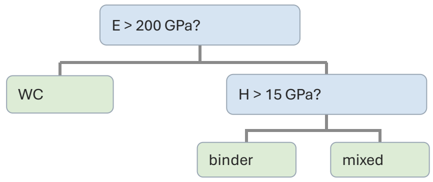
Decision tree
- A flowchart of **yes/no splits** on the features (is modulus > 200 GPa?); each path ends at a leaf that assigns a phase

Random forest (bagging)
- Grow **many trees**, each on a bootstrap resample and a random subset of features; **average their votes** so independent errors cancel (low variance)

Boosting (XGBoost)
- Add trees **in sequence**, each correcting the last one's errors (low bias)

HDBSCAN (clustering)
- Groups points in **dense regions**; no preset k, sparse points labelled noise (−1)

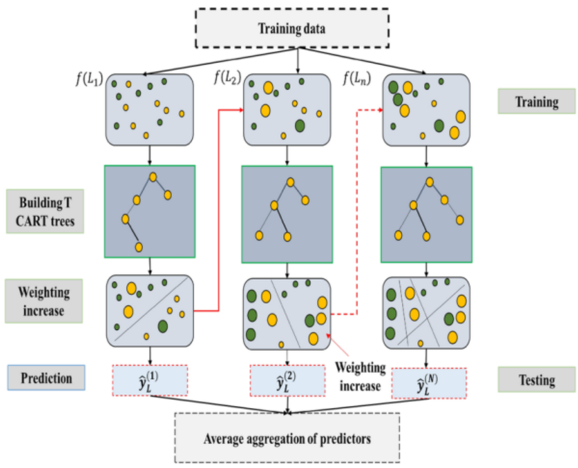




A decision tree: yes/no splits down to a leaf. A random forest = many such trees, majority vote.

XGBoost: Gradient Boosted Decision Trees

- XGBoost (Extreme Gradient Boosting) is a high-performance implementation of **gradient-boosted decision trees**, optimized for speed, accuracy, and interpretability.
- Build an ensemble of **weak learners** (decision trees) where each new tree **corrects the residuals** (errors) of the previous ensemble.



The rare-phase problem: SMOTE

The rare-phase trap

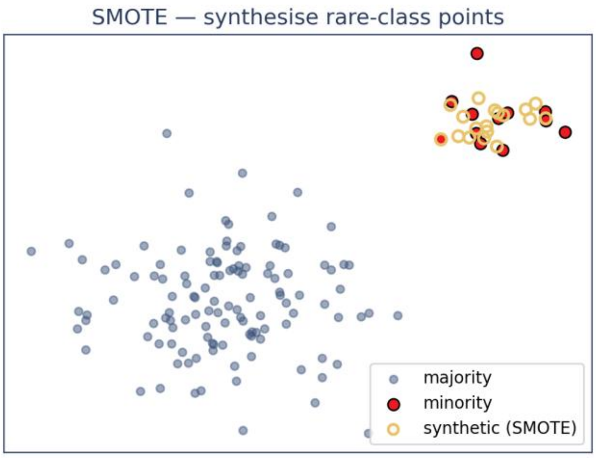
- A minority phase is a few percent of indents, so a model can score high **accuracy** while never predicting it: judge by **recall** (fraction of the rare class actually caught), not accuracy

SMOTE

- **Synthetic Minority Over-sampling**: create new minority examples on the line between a real one and a neighbour, $x_{new} = x_i + \lambda(x_{nn} - x_i)$, $\lambda \in [0,1]$
- Adds variety instead of duplicating rows, so the classifier sees a balanced training set

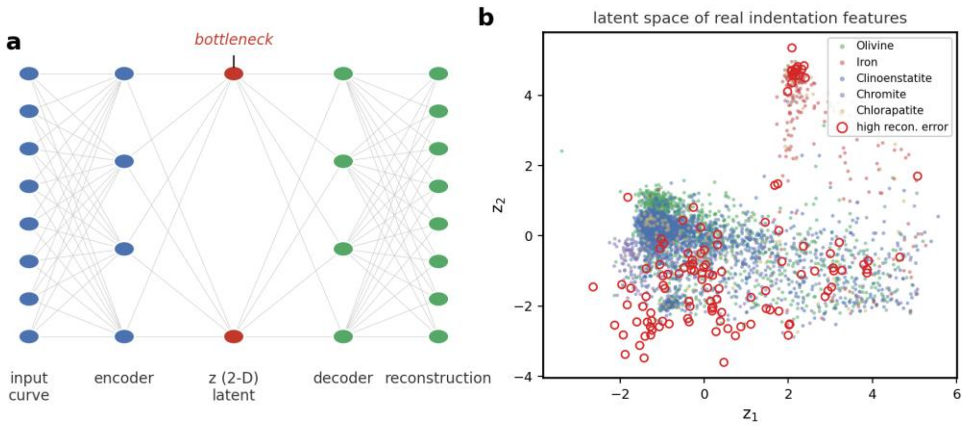
Here

- Used in the meteorite label refinement (XGBoost + SMOTE)



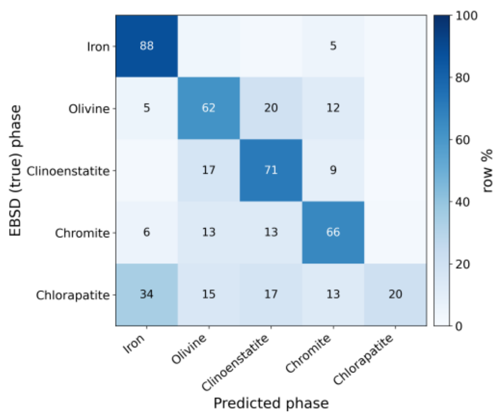
Autoencoders and the latent space

Autoencoder: a network trained to reproduce its input through a narrow **bottleneck**, so it must compress each load-depth curve into a few numbers (the **latent space**) and rebuild it. Similar curves land close together in that space, giving an unsupervised map that needs no labels.



Trusting a model: evaluation & validation

MecaNano



Cross-validation

- Rotate the held-out fold K times; report a **mean ± std**, not one lucky split

Confusion matrix (left)

- Rows = true phase, columns = predicted: the **diagonal is correct**, off-diagonal are the mistakes

Precision / recall / F1

- **Precision**: of those called X, how many are X = TP/(TP+FP)
- **Recall** (true-positive rate): of the real X, how many caught = TP/(TP+FN)
- **F1**: harmonic mean of precision and recall

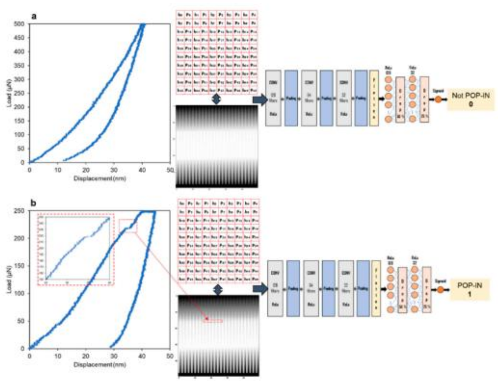
ROC / AUC

- ROC: **true-positive rate vs false-positive rate** (FP/(FP+TN)) as the threshold slides
- **AUC**: area under the curve = 1 perfect, 0.5 chance

Detecting anomalies: pop-in events

MecaNano

Unpublished data



Pop-ins

- Sudden **displacement bursts** on the load-depth curve, the signature of dislocation nucleation or film/crack events under the tip

Why it matters

- They break the smooth Oliver-Pharr fit, so an undetected pop-in corrupts the extracted **H and Er** across 10^5 auto-fitted curves

CNN detection

- **Convolutional neural network**: slides small learned filters over the curve to detect the local burst pattern, separating typical from anomalous indents (~93%)

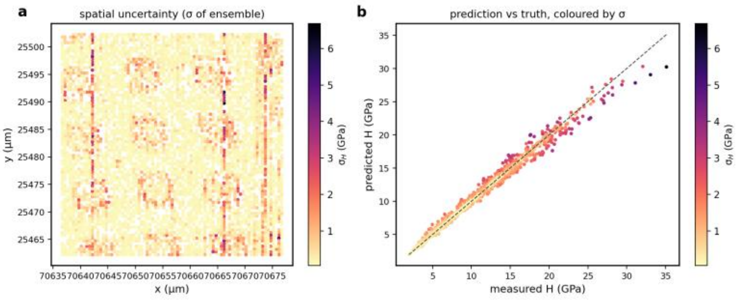
A. Daugela, C.H. Chang, D.W. Peterson, Mater. Sci. Eng. A 800, 140273 (2021).

How sure is the model?: uncertainty

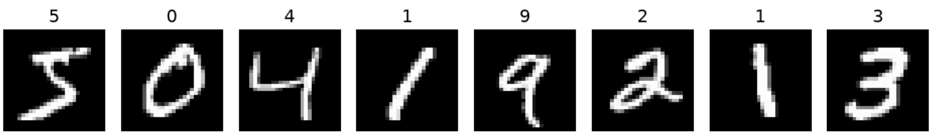
Dropout
- During training, randomly **switch off** a fraction of neurons each pass, so the network cannot over-rely on any one unit (a regulariser against overfitting)

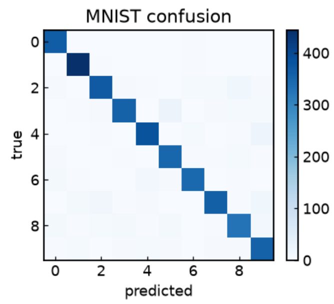
MC-dropout
- Keep dropout **on at prediction** and run the input ~50 times: each pass is a different sub-network, so the **spread** of the outputs is the uncertainty

Reading it
- σ peaks at grain boundaries and phase mixtures, flagging where to measure next



A gentle warm-up: teaching a CNN to read digits





Before turning to nanoindentation, it helps to watch a convolutional neural network (CNN) work on a problem everyone recognises: reading handwritten digits. MNIST is the classic benchmark: 70,000 small greyscale images of the numbers 0–9.

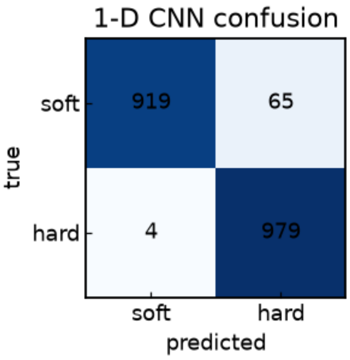
A CNN slides small learnable filters across each image to pick out strokes and curves, stacks them into higher-level shapes, and outputs a probability for each digit. After only a couple of minutes of training on a laptop CPU it reaches about **99% accuracy**: the confusion matrix is almost perfectly diagonal, so nearly every digit is read correctly.

The very same machinery: filters, pooling, a probability output: is what later reads indentation curves and micrographs. MNIST is just the friendliest place to meet it.

▶ Run it: notebook 10

scan or click · Colab, no install

Two ways to feed a CNN: the raw curve itself (1-D)



A load–depth curve is just a sequence of numbers, so a CNN can read it **directly**: a one-dimensional filter slides along the curve looking for characteristic shapes: the loading slope, the unloading knee, a pop-in step.

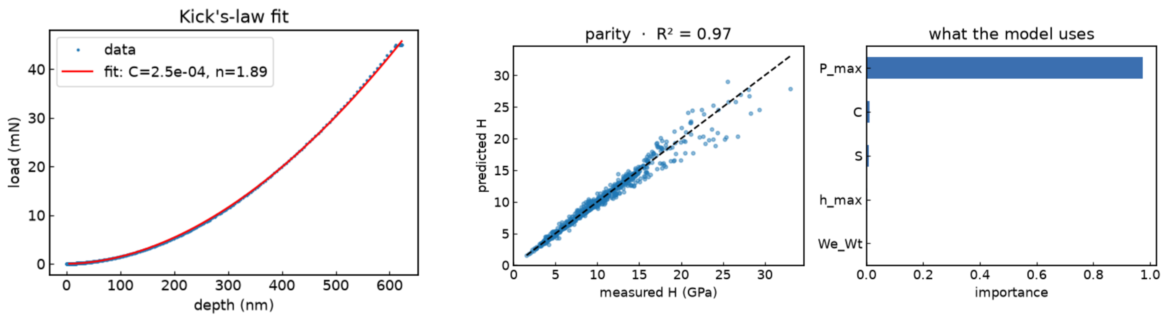
This is the counterpart to notebook 04, which first turned the curve into a 2-D image. The 1-D version keeps the true **magnitude** of load and depth; the image version emphasises **shape** and is scale-invariant. Choosing between them is a real modelling decision, and the tutorial shows both so you can compare.

Trained to tell soft from hard indents, the 1-D CNN reaches about **96% accuracy** (confusion matrix: only 69 of ~2,000 curves misclassified).

▶ Run it: notebook 04a

scan or click · Colab, no install

Predicting numbers: physics fits and regression



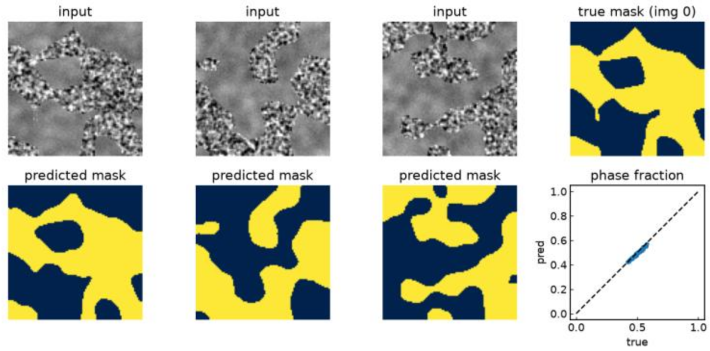
Not every task is classification: sometimes we want a continuous number. Two complementary routes are shown. A **physics model**: Kick's law $P = C{\cdot}h^n$ is fitted to the curve (left), recovering interpretable constants (here $C = 2.5{\times}10^{-4}$ and exponent $n = 1.89 \approx 2$, the value expected for a self-similar indenter). A **data-driven model**: a random-forest regressor predicts hardness from simple curve descriptors and reaches $R^2 = 0.97$ against the measurements (parity plot). The bar chart shows what it relies on: the maximum load P_max dominates: so the prediction is interpretable, not a black box.

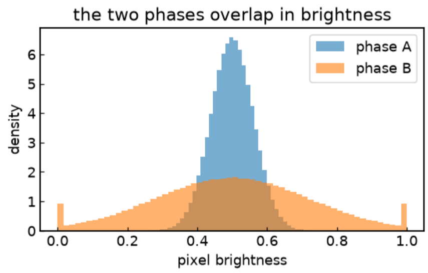
▶ Run it: notebook 12

scan or click · Colab, no install

Labelling **every pixel**: U-Net image segmentation

MecaNano





Clustering labels each indent; **segmentation labels each pixel** of an image. A U-Net is a convolutional network shaped like a U: it compresses the image to grasp context, then expands it back to full resolution, so it can outline phases or defects precisely. It is needed because the two phases **overlap in brightness** (histogram): a simple grey-level threshold fails, but the network learns their texture and shape. Trained on a handful of annotated micrographs it predicts masks that match the ground truth, and the predicted phase fraction falls on the parity line: the route from raw images to quantitative phase maps.

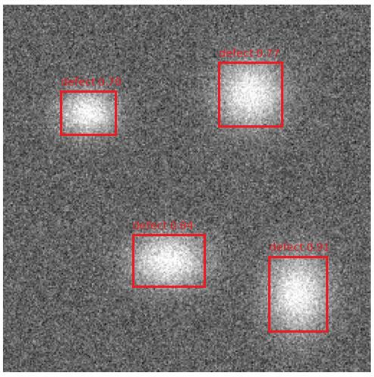
▶ Run it: notebook 15

scan or click · Colab, no install

Finding and counting features: **object detection**

MecaNano

illustrative detections (box · class · confidence)



Some questions are not 'what phase is this?' but **'where are the features, and how many?'**: counting precipitates, pores or etch-pits across a micrograph.

Object-detection networks such as **YOLO** ('You Only Look Once') answer this in a single pass: they draw a bounding box around each feature and attach a class and a confidence score. Because it localises and counts automatically, it replaces hours of manual annotation and gives reproducible statistics: density, size distribution, spatial clustering: over large images.

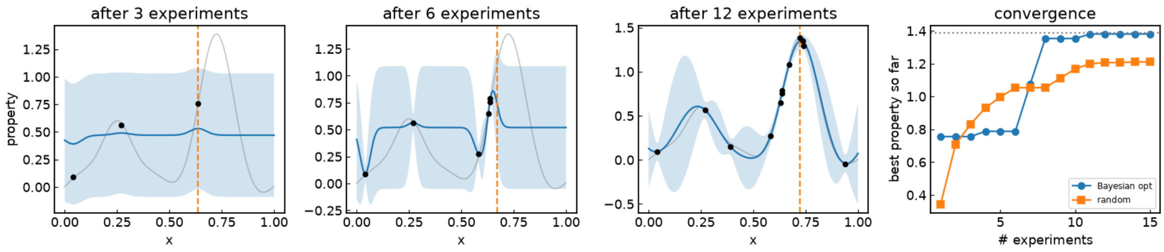
Shown here on synthetic micrographs so it runs anywhere; the same network retrains on real defect images. It is the most advanced, optional tool in the set.

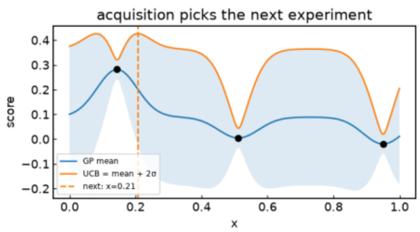
▶ Run it: notebook 13

scan or click · Colab, no install

Choosing the next experiment: Bayesian optimisation

MecaNano





Bayesian optimisation

- Update belief with data (posterior ∝ likelihood × prior); each result chooses the next experiment, so few runs reach the best recipe
- **Surrogate** (Gaussian process): the mean prediction with its ±uncertainty band (the shaded band, wide where data is sparse)
- **Acquisition**: samples next where payoff or uncertainty is highest (exploit vs explore)

▶ Run it: notebook 16

scan or click · Colab, no install

33

Run along: methods and the notebooks

MecaNano

- Foundations & PCA · **00, 01, 01a**
- Clustering: k-means, GMM, HDBSCAN · **02**
- Nearest neighbours (k-NN) · **02a**
- Trees / RF / boosting, SHAP, SMOTE · **03**
- Model evaluation: CV, ROC · **03a**
- Curve→image CNN · 1-D CNN · **04, 04a**
- Autoencoder & latent space · **05**
- Correlative registration · **06**
- Substrate / layer deconvolution · **07**
- Depth-resolved mapping · **08**
- Multimodal pipeline · **09**
- Pop-in detection · **11**
- Curve fitting / RF regression · **12**
- Uncertainty quantification · **14**
- Segmentation · Bayesian optimisation · **15, 16**

scan → Binder

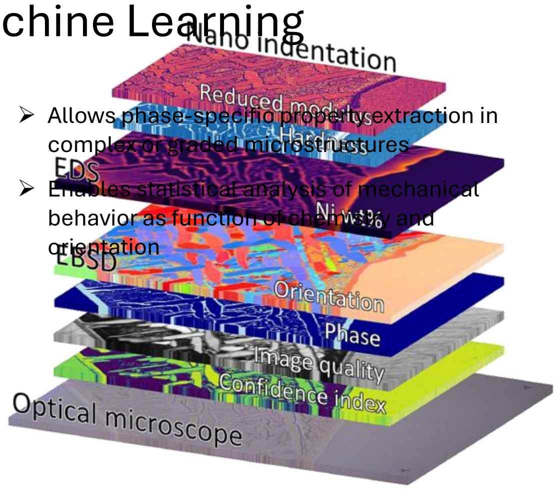
34

Why correlative characterization?

- ➢ Combines mechanical maps (nanoindentation) with compositional and structural maps (EDS, EBSD)
- ➢ Purpose: link local mechanical properties to phase, composition, and crystal orientation
- ➢ Applied in diverse systems: alloys, composites, natural materials, meteorites

Machine Learning

- ➢ Allows phase-specific property extraction in complex or graded microstructures
- ➢ Enables statistical analysis of mechanical behavior as function of chemistry and orientation



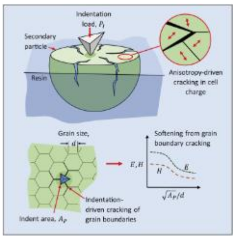
35

How can I use this knowledge for materials characterization?

Li-ion cathodes

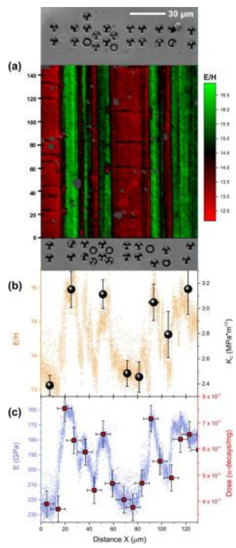
- ➢ Only co-registered mechanical, structural, and chemical information can reveal the true degradation drivers.

https://doi.org/10.1016/j.joule.2022.04.001



Ion-irradiated materials

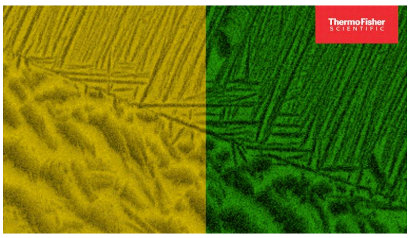
- ➢ local mechanical response with defect/chemistry maps is required to build and validate hardening and embrittlement models.



Tobias Beirau, et. al. *Appl. Phys. Lett.* 6 December 2021; 119 (23): 231903. https://doi.org/10.1063/5.0070597

TRIP/TWIP steels

- ➢ Measure mechanics and phase/defect evolution on the same microstructural units to understand and control strength–ductility



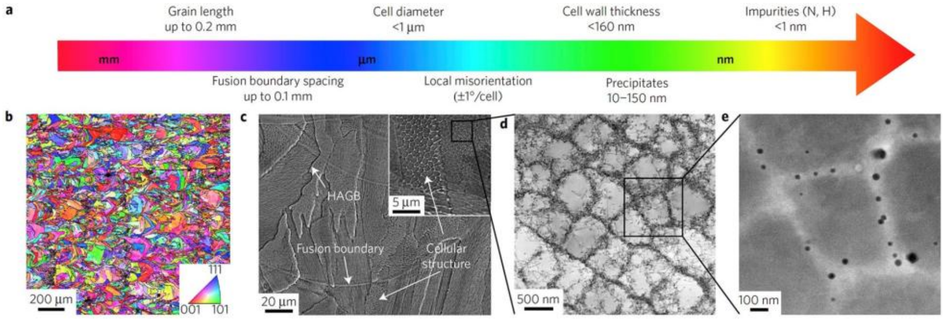
36

Establishing correlations

Roma Tre

Machine learning allows building
structure-property-processing relationships across length scales.

Laser powder-bed-fusion (L-PBF) produced 316L stainless steel (SS)



We need (e.g.):

1. EBSD
2. SEM
3. TEM
4. APT
5. Nanoindentation
6. EDS

Wang, Y., Voisin, T., McKeown, J. *et al.* Additively manufactured hierarchical stainless steels with high strength and ductility. *Nature Mater* **17**, 63–71 (2018). https://doi.org/10.1038/nmat5021

37

Roma Tre

Shifting gears

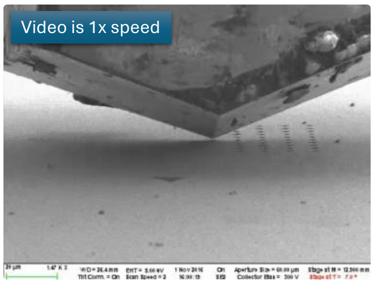
Applications to high-speed nanoindentation

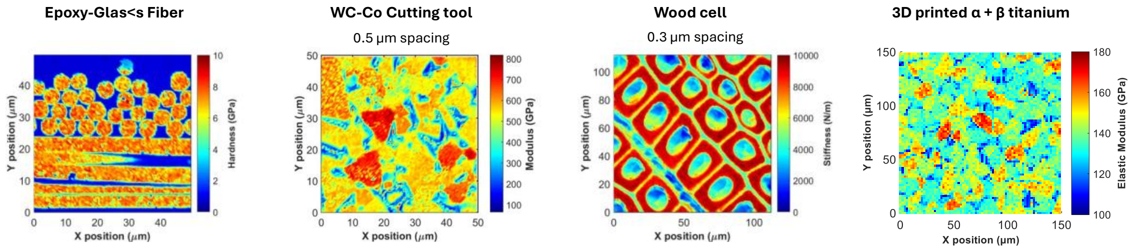
38

Roma Tre

High-speed (high-resolution) nanoindentation

- What does it take to do high-speed nanoindentation?
  - Compression in time (normal way, but do it fast)
  - One full load-unload cycle and movement to the next position can be completed in less than one second
- Why should you do it?
  - High-throughput data (statistics)
  - Probe heterogeneity (mapping)
  - More likely to be accurate (thermal transients, vibrations, environment). Less influence of the environment.





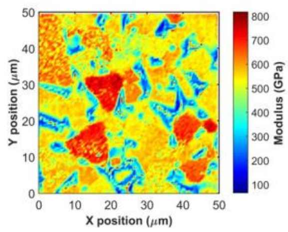
Roma Tre

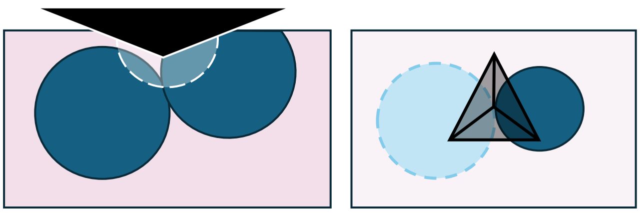
How do I deconvolute into useful information?

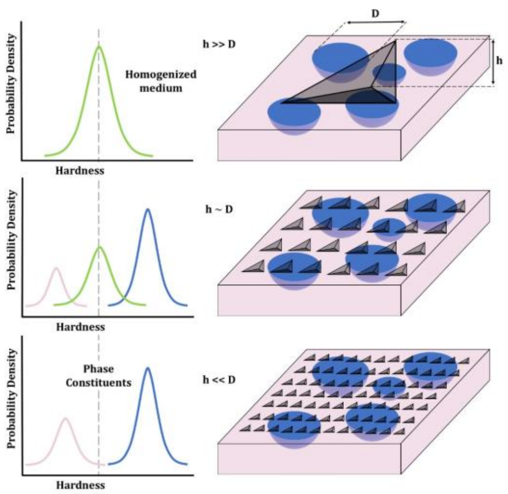
Properties are convoluted

What influences phase deconvolution?

- Relation D (lateral indent size) & h (depth)
- Sub-surface features
- Surroundings

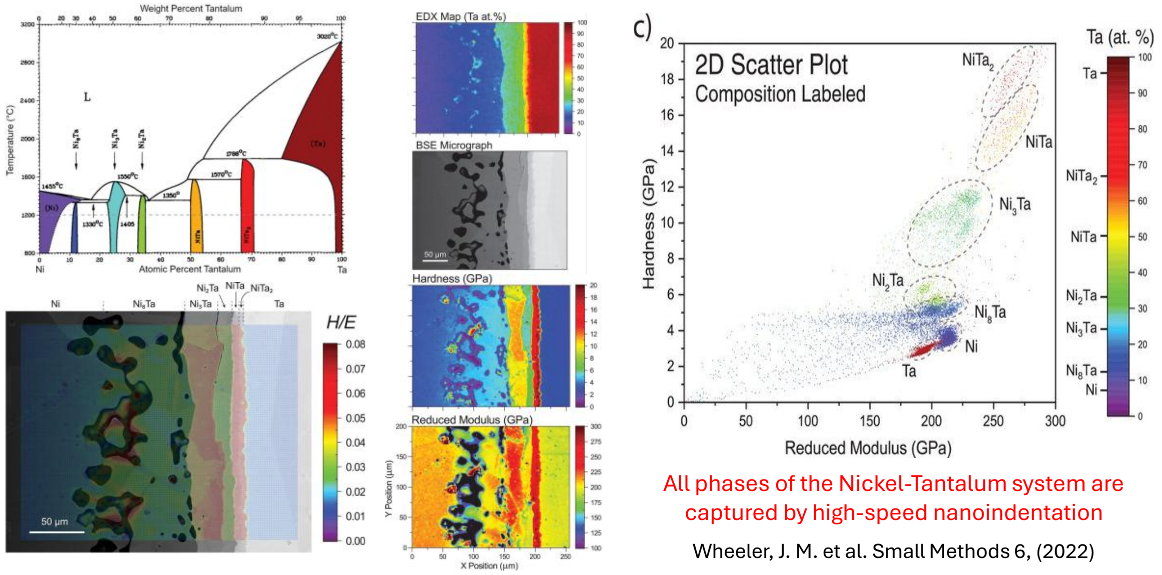






Roma Tre

Traditional Correlative characterization: EDS, Nanoindentation



All phases of the Nickel-Tantalum system are captured by high-speed nanoindentation

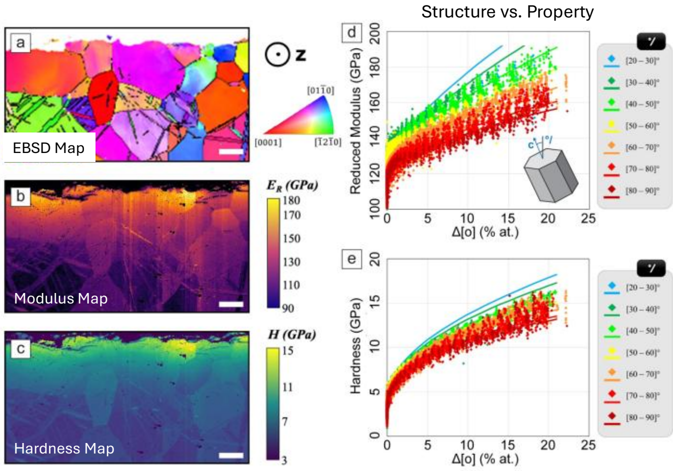
Wheeler, J. M. et al. Small Methods 6, (2022)

Roma Tre

Traditional correlative characterization: EPMA, EBSD, Nanoindentation

- Combined EBSD, EPMA, and nanoindentation reveal the role of:
  - Crystal orientation on modulus and hardness.
  - Oxygen content on mechanical strengthening.
- Reduced modulus and hardness show orientation-dependent trends.
- Data correlation allows separating anisotropy effects from compositional variations.
- Approach applicable to high-throughput studies of alloy development.

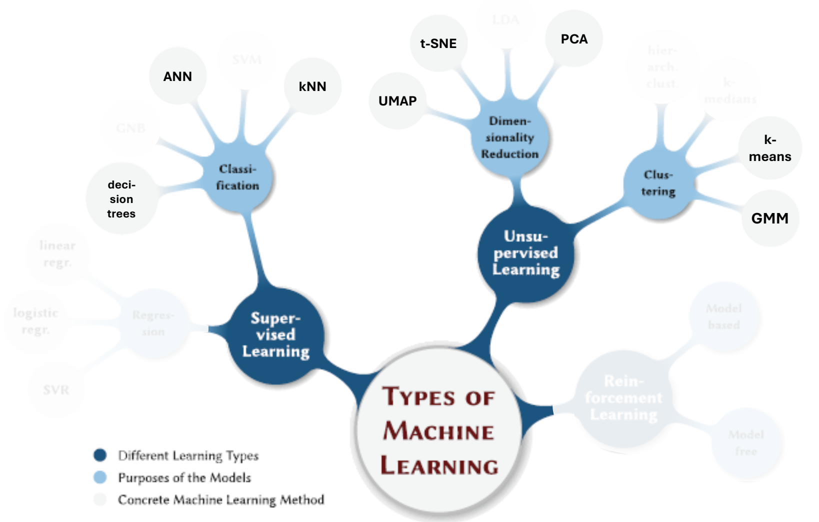
Commercially pure titanium



Correlated nanoindentation, EBSD, and EPMA on commercially pure titanium. Mechanical properties (modulus and hardness) are linked to grain orientation and oxygen concentration.

Dealing with complexities: data-driven approach

Roma Tre

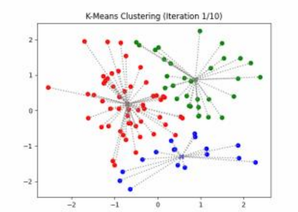


S. Sandfeld, The "Materials Data Science" book (2024, Springer)

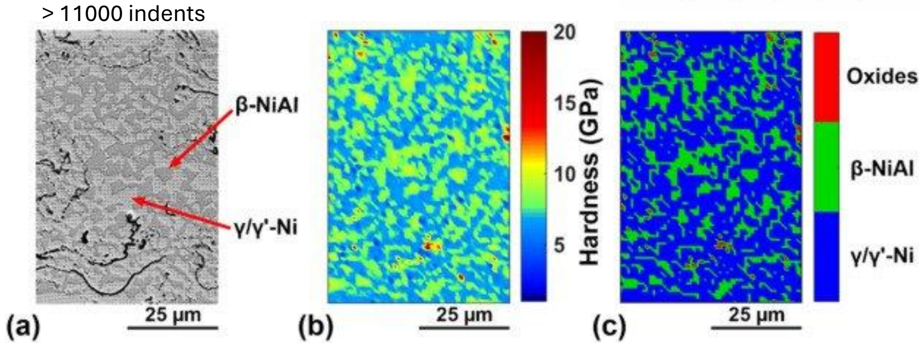
Simple multiphase alloy: ML based clustering

Roma Tre

- Unsupervised Clustering: Groups data based on similarity without labels.
- K-Means, k-medoids Clustering:
  - Divides data into k clusters.
  - Steps: Initialize centroids, assign points, update centroids, repeat until stable.



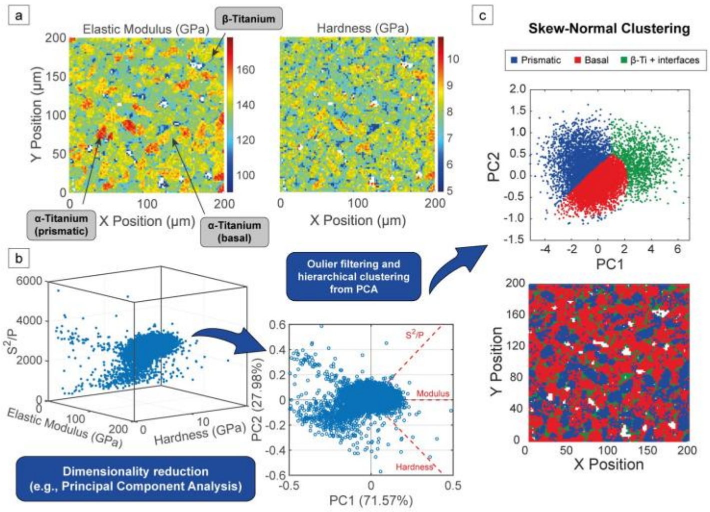
> 11000 indents



Vignesh, B., et al. "Critical assessment of high speed nanoindentation mapping technique and data deconvolution on thermal barrier coatings." *Materials & Design* 181 (2019): 108084.

Adding complexity: hyperspace ML clustering

- Gaussian Mixture Modeling (GMM) and Skew Normal Mixture Modeling
  - Fits data as a combination of Gaussian/skew normal distributions.
  - Effective for separating mixed data into distinct subpopulations.
  - 2D GMM/SKMM: Commonly used for clustering and density estimation in 2D datasets



E. Rossi, C. Tromas, Z. Liu, Y. Zou, and J. M. Wheeler, "Revealing new depths of information with indentation mapping of microstructures," *MRS Bulletin*, accepted for publication, Apr. 2025.

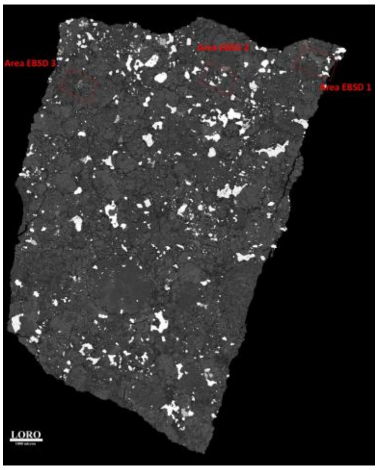
LORO meteorite

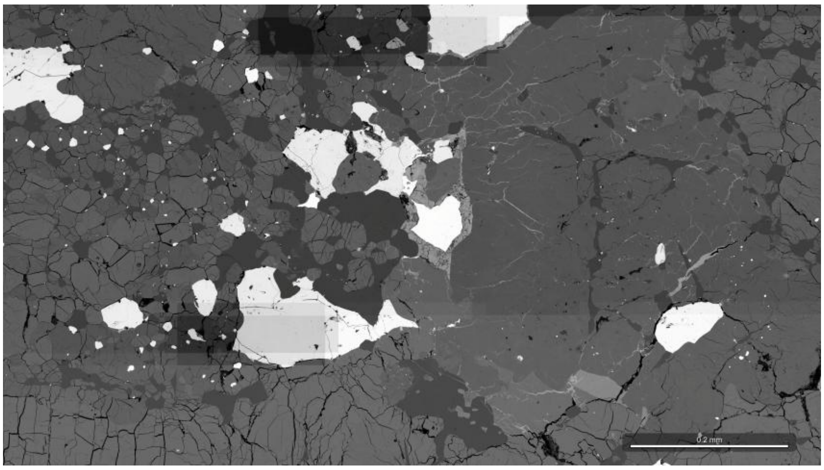
All complexities together!

Complexity in microstructure

Roma Tre

The Loro meteorite is a fresh ordinary chondrite (L6) composed primarily of Mg–Fe silicates, with minor Fe–Ni metal and sulfides.

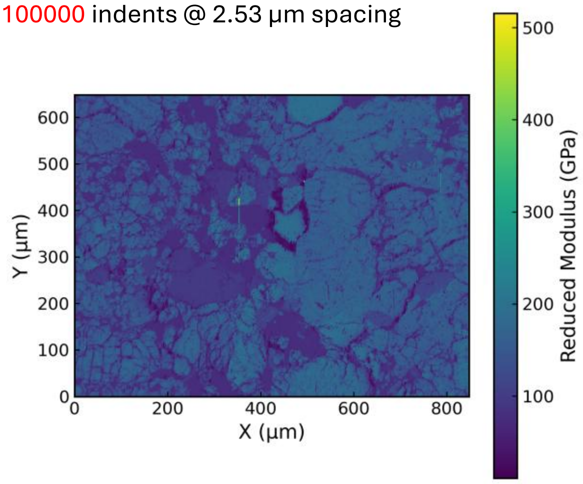


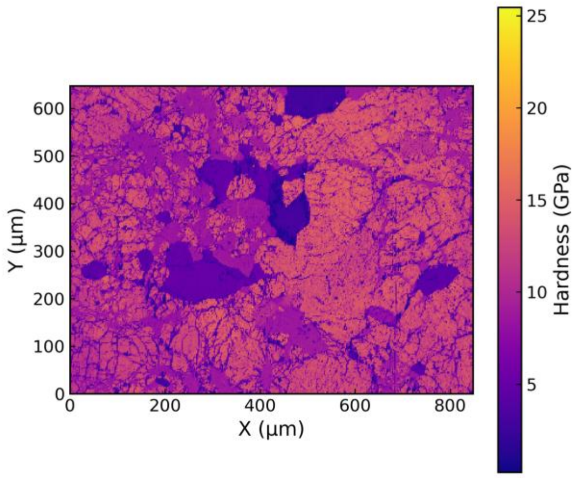


High-speed nanoindentation maps

Roma Tre

100000 indents @ 2.53 μm spacing

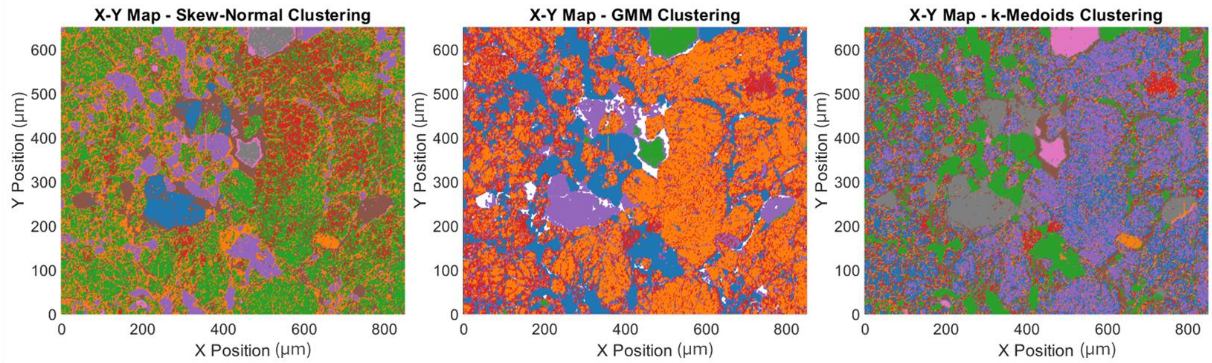




Roma Tre

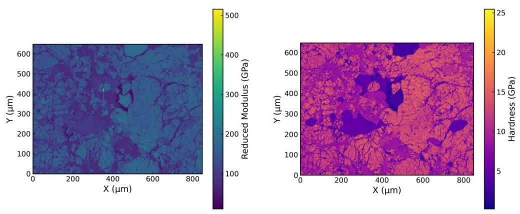
Simple first step: unsupervised clustering

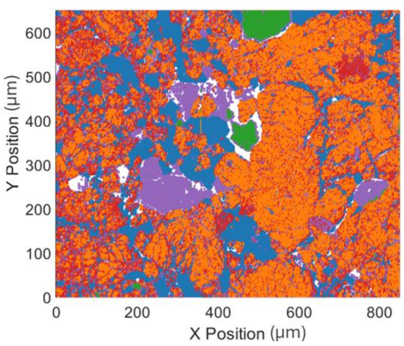
- Targeted 5 mechanical phases.
- Different colors represent regions of homogeneous mechanical properties as identified by ML clustering techniques.



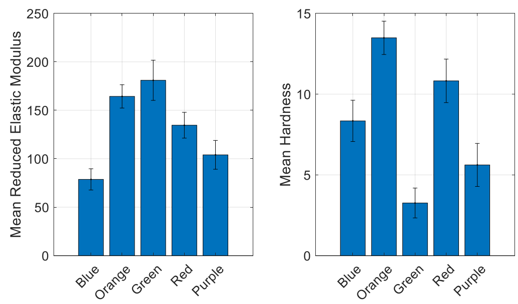
Roma Tre

Simple first step: unsupervised clustering





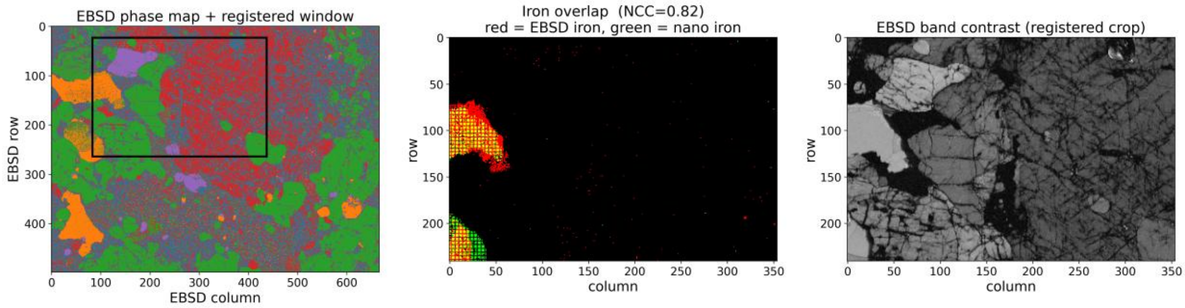
Cluster Color	R. Elastic Modulus (GPa)	Mean Hardness (GPa)
Green		
Blu		
Orange		
Purple		
Red		

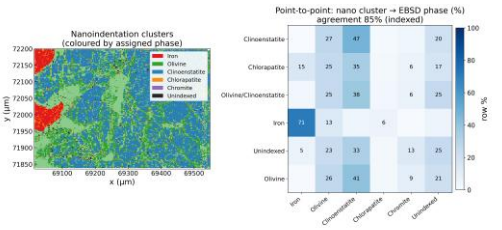


Correlation for learning: EBSD registration

Roma Tre

Unpublished data





Affine EBSD → map registration aligns the two modalities to **NCC = 0.82**, correcting tilt, drift and scale so each indent overlays the correct EBSD pixel.

Point-to-point, **85% of indexed indents** then agree with the mapped EBSD phase: trustworthy enough to transfer crystallographic labels onto the mechanical map and train on them, without re-measuring EBSD.

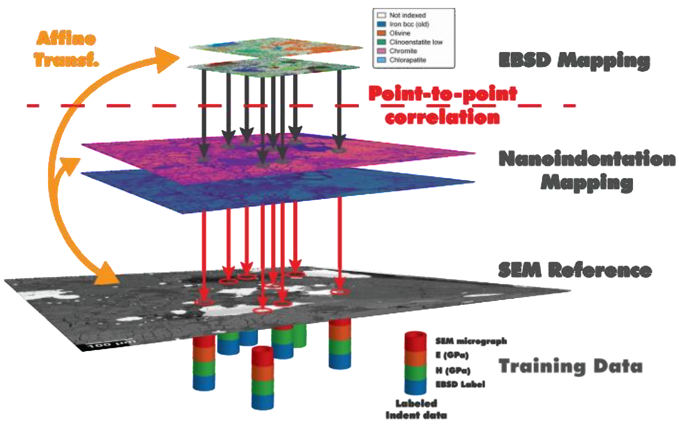
M. Seehaus, C.H. Lee, T. Stollenwerk, J.M. Wheeler and S. Korte-Kerzel: *Materials & Design* **234**, 112296 (2023).

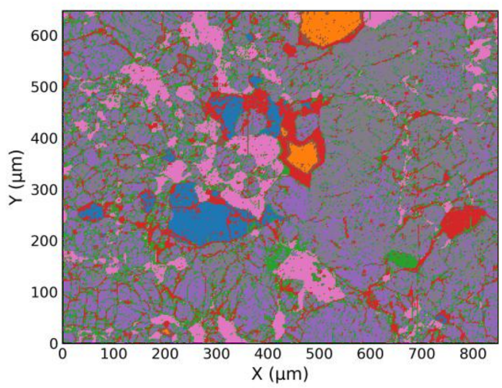
Supervised clustering for label refinement

Roma Tre

Unpublished data

The workflow





Roma Tre

Deep learning applied to nanoindentation curves

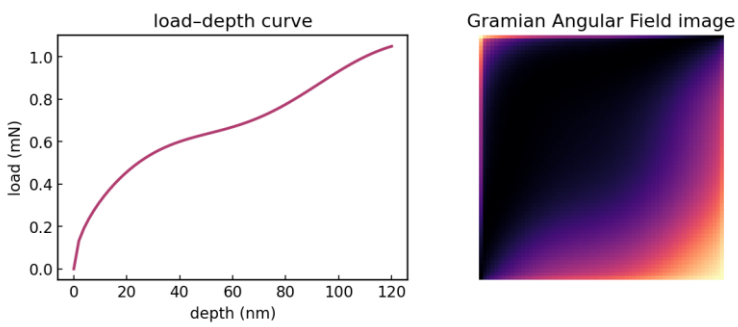
Machine learning to deconvolute phases in hard metals

L. Ortiz, E. Rossi, et al. (unpublished)

56

Roma Tre

The curve becomes an image: Gramian Angular Field



Curve → image rescale the curve to angles, then encode pairwise sums as a 2-D map (GAF).

Why a CNN can now read the whole load–depth response: the basis of the WC–Co classifier.

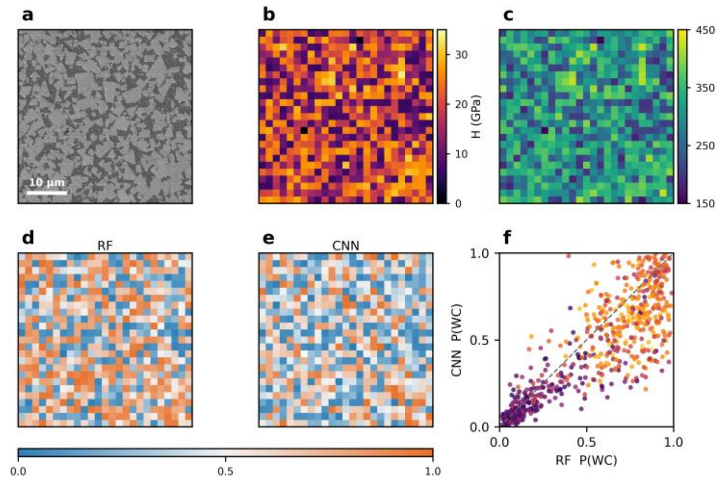
Caveat the GAF is scale-invariant: it sees shape, not magnitude · notebook 04.

Laia Ortiz, Edoardo Rossi, Claus Trost et al. (unpublished)

60

Probabilistic phase maps agree

Roma Tre



The hardness and modulus maps, and the RF and CNN probability maps, all **track each other** (panel f): two independent classifiers reaching the same answer is strong evidence the maps are real, not artefacts.

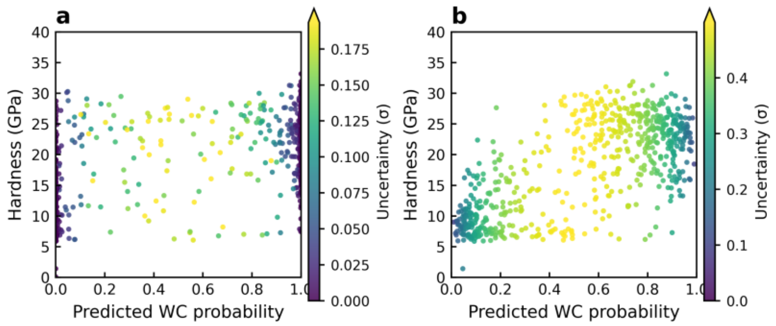
Because the estimate is statistical, **binder pockets finer than the indent spacing** survive as elevated binder probability rather than being averaged away.

Laia Ortiz, Edoardo Rossi, Claus Trost et al. (unpublished)

61

How sure is the model? Prediction uncertainty

Roma Tre



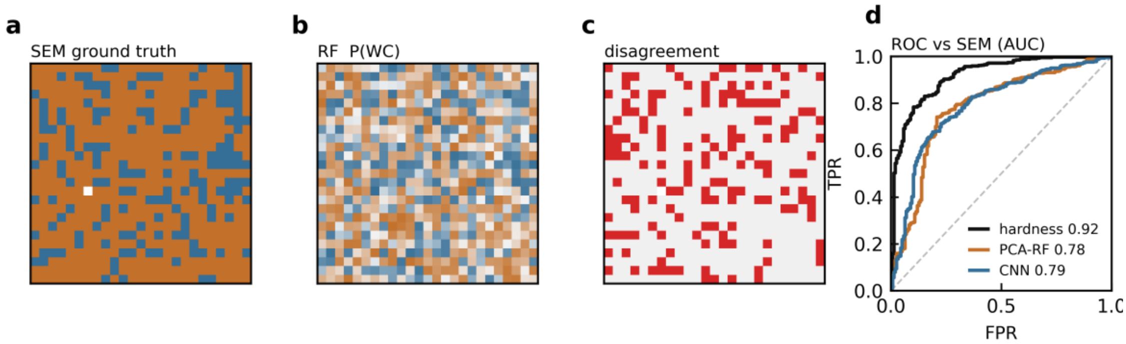
Prediction uncertainty peaks at **P(WC) ≈ 0.5** (the genuinely mixed indents where WC and binder overlap) and collapses toward pure WC or pure binder. The model does not just classify; it reports where to trust the map and where the microstructure itself is ambiguous.

Laia Ortiz, Edoardo Rossi, Claus Trost et al. (unpublished)

62

Validation against SEM ground truth

Roma Tre



Against SEM segmentation, ROC-AUC = **0.92 (hardness), 0.78 (PCA+RF), 0.79 (CNN)**. Disagreement concentrates at WC/binder boundaries: the same mixed indents flagged as uncertain, so errors are honest and interpretable.

Laia Ortiz, Edoardo Rossi, Claus Trost et al. (unpublished)

64MCTF 2026

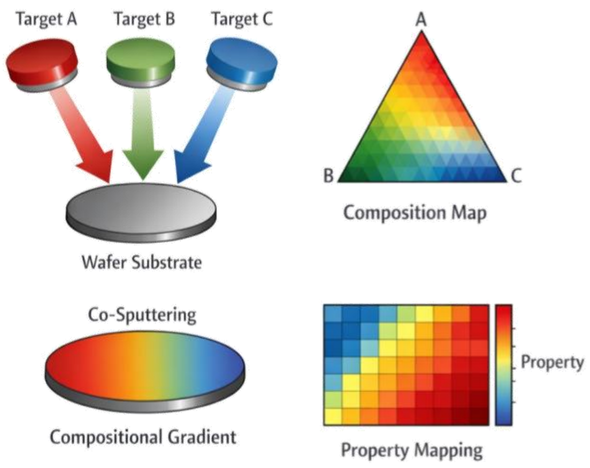
Roma Tre Montanuniversität Leoben

Correlative mechanical microscopy to extract structure property correlations

combination of HSNM with EDS, TEM and micro-pillar testing

Edoardo Rossi et al. (unpublished)

Combinatorial PVD: Accelerating the Discovery of Complex / Metastable Alloy Thin Films



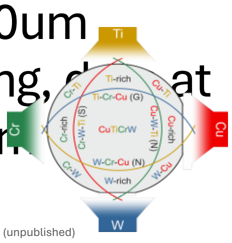
- Efficient exploration of high-dimensional compositional spaces (e.g., HEAs, multi-principal element alloys)
- Access to non-equilibrium phases and metastable structures not achievable in bulk
- Ideal platform for high-throughput structure–property mapping and correlative probing (EDX, XRD, nanoindentation)

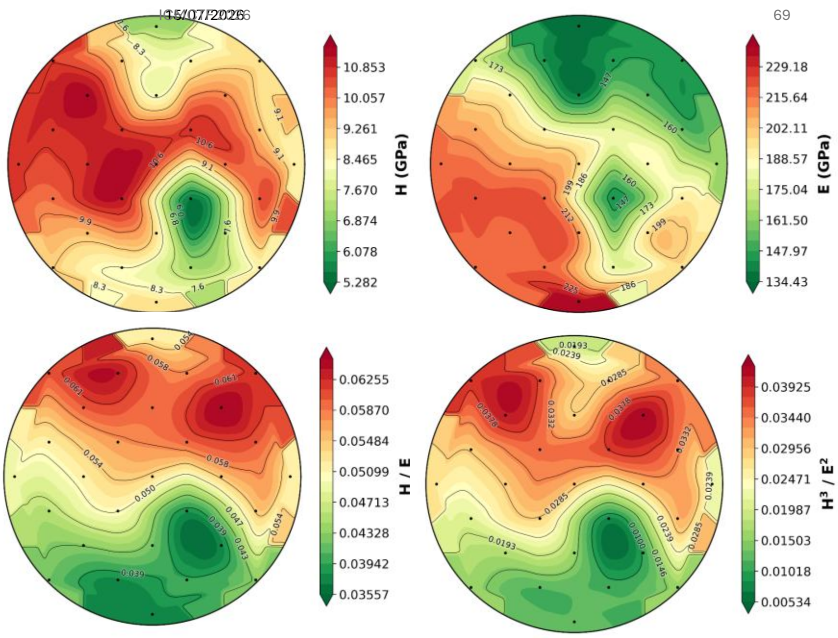
Edoardo Rossi et al. (unpublished)

https://doi.org/10.1016/j.matdes.2025.113881

High-speed nanoindentation mapping of the wafer

3D Slice, 30 indentations 6x5 10um spacing, d... at 120 nm

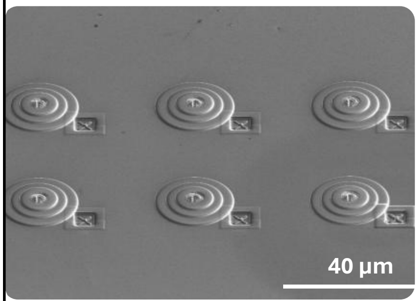


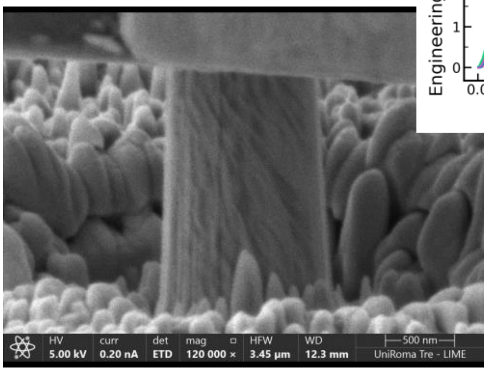


Edoardo Rossi et al. (unpublished)

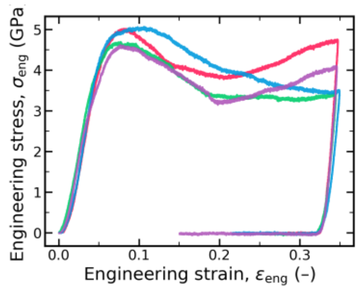
In-situ micro pillars compression mapping

- FIB milled micro pillars with 1µm radius, and 2 µm height, depending on the film thickness
- In-situ compression with flat punch tip





16X speed up



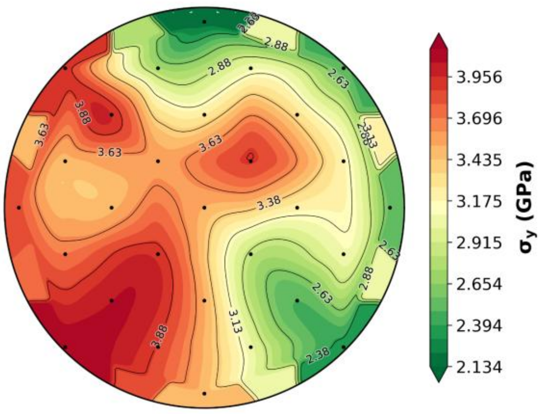
- Stress-strain curves to evaluate the mechanical response during compression

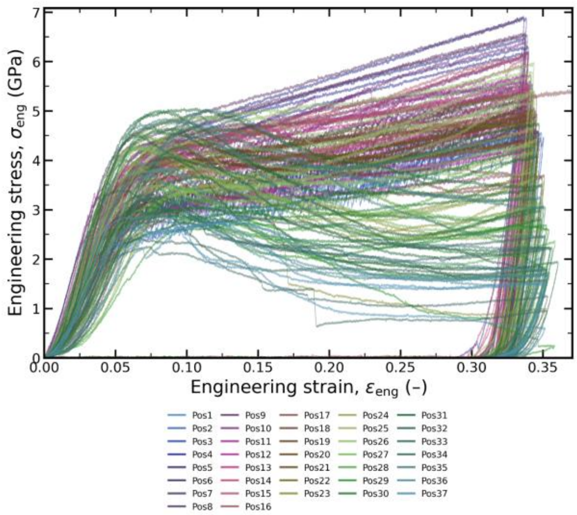
Edoardo Rossi et al. (unpublished)

Roma Tre Montanuniversität Leoben

Pillar-compression mapping (5 pillars x POI)

Yield Strength (GPa)



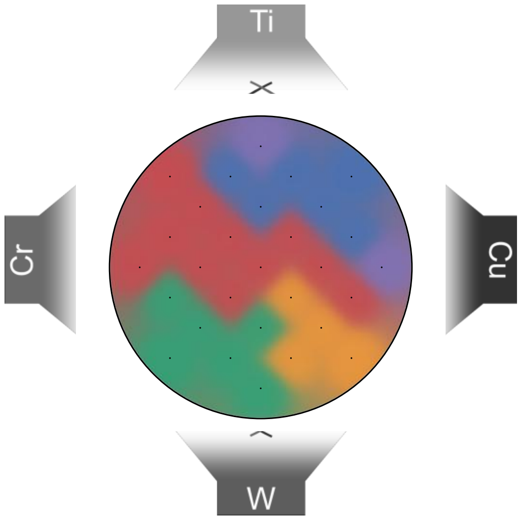


Edoardo Rossi et al. (unpublished)

Roma Tre Montanuniversität Leoben

How can we identify automatically? ML clustering

- ➢Multi-scale clustering combining nanoindentation (H, E) and micropillar compression (σ_y, σ_{max}/σ_y) with derived ratios (H/E, H^3/E^2): 6 features, z-score standardised
- ➢Ward hierarchical clustering with silhouette-driven k selection; k=4 optimal (silhouette = 0.41, perfect Ward–KMeans agreement ARI = 1.0)
- ➢**4 mechanically distinct regions plus 1 outlier region**



Edoardo Rossi et al. (unpublished)

Roma Tre Montanuniversität Leoben

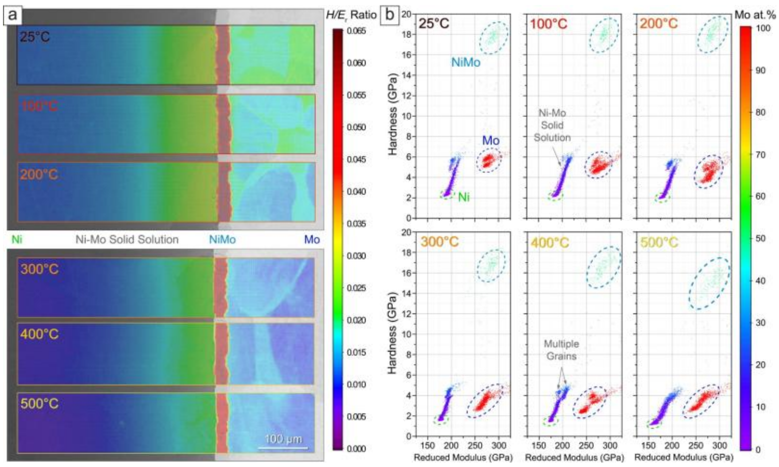
72

Towards Operando and Fully Automated Mapping

Roma Tre

Operando: measuring the mechanical properties of materials while they are exposed to service-like conditions during the experiment.

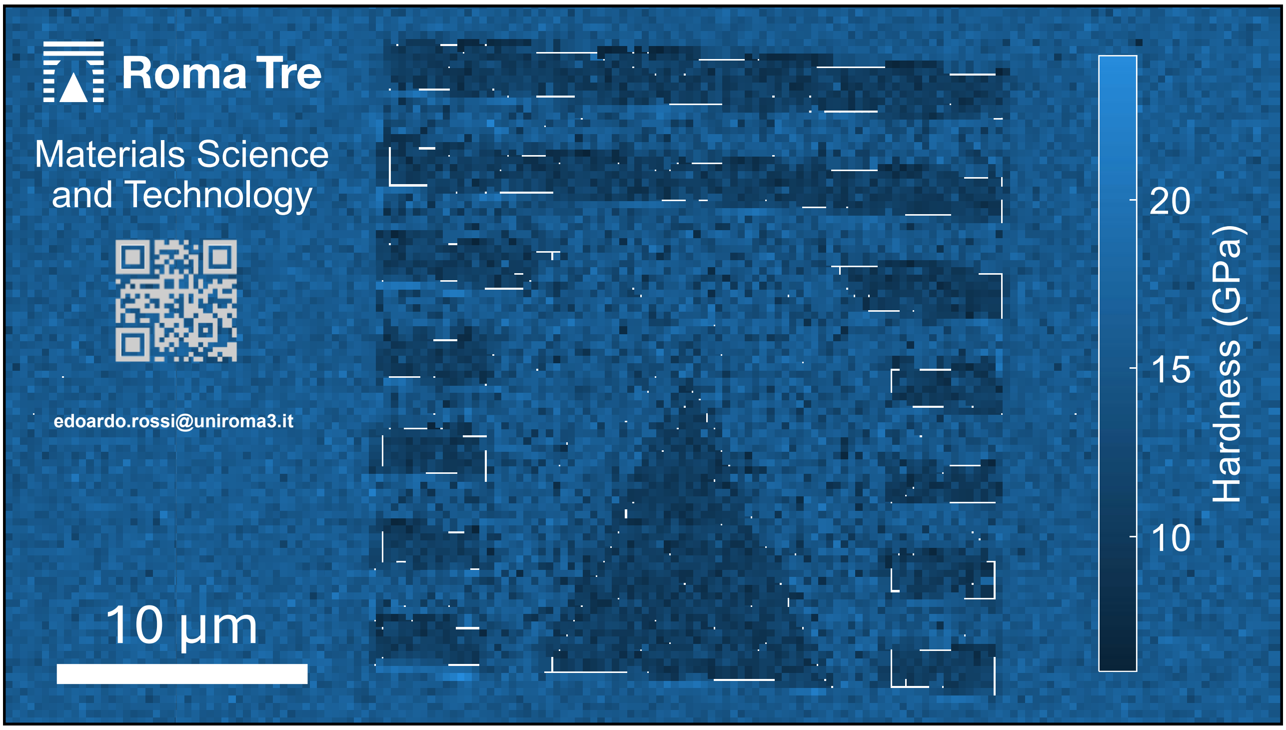
- ➢ Operando mapping extends high-speed nanoindentation to temperature- and environment-controlled experiments.
- ➢ Simultaneous correlation with compositional mapping (EDS) enables:
  - ➢ Phase boundary tracking
  - ➢ Diffusion profile characterization
- ➢ Real-time ML-driven adaptive mapping adjusts resolution and sampling dynamically.
- ➢ Towards fully autonomous mechanical microscopy platforms.



Ni-Mo diffusion couple at temperature.

7/15/2026 Edoardo Rossi - MecaNano WG4 – Tutorial 1 76

76


Roma Tre
Materials Science
and Technology
edoardo.rossi@uniroma3.it
10 µm
20
15
10
Hardness (GPa)

77