THE UNIVERSITY of TENNESSEE KNOXVILLE

# Deep Kernel Learning - I

Sergei V. Kalinin

# What have we learned from lectures on GP/BO

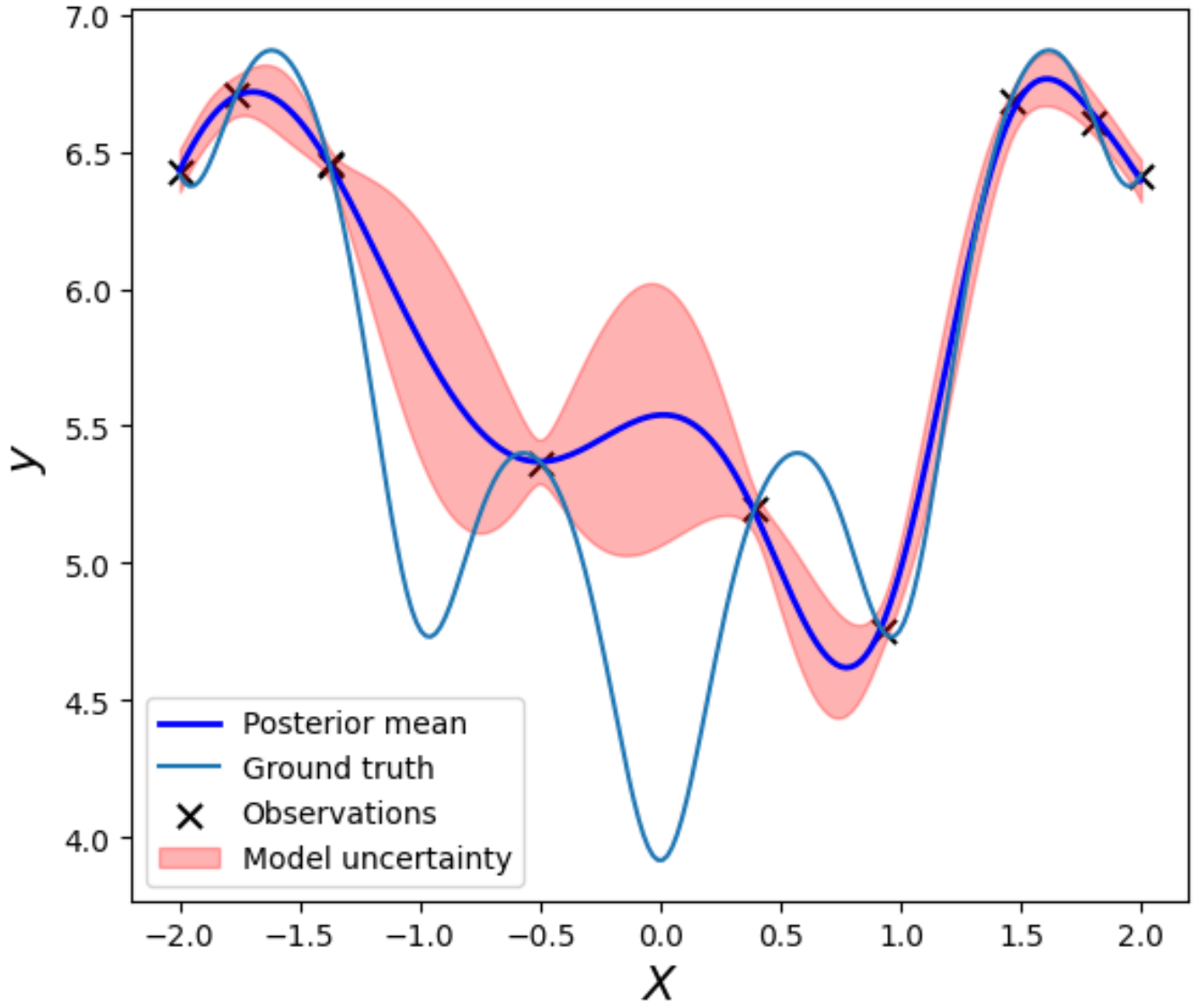
- Gaussian Process

- Kernel and kernel parameters
- Kernel Priors
- Noise Priors
- Mean function and priors

- Posteriors
- Bayesian Inference

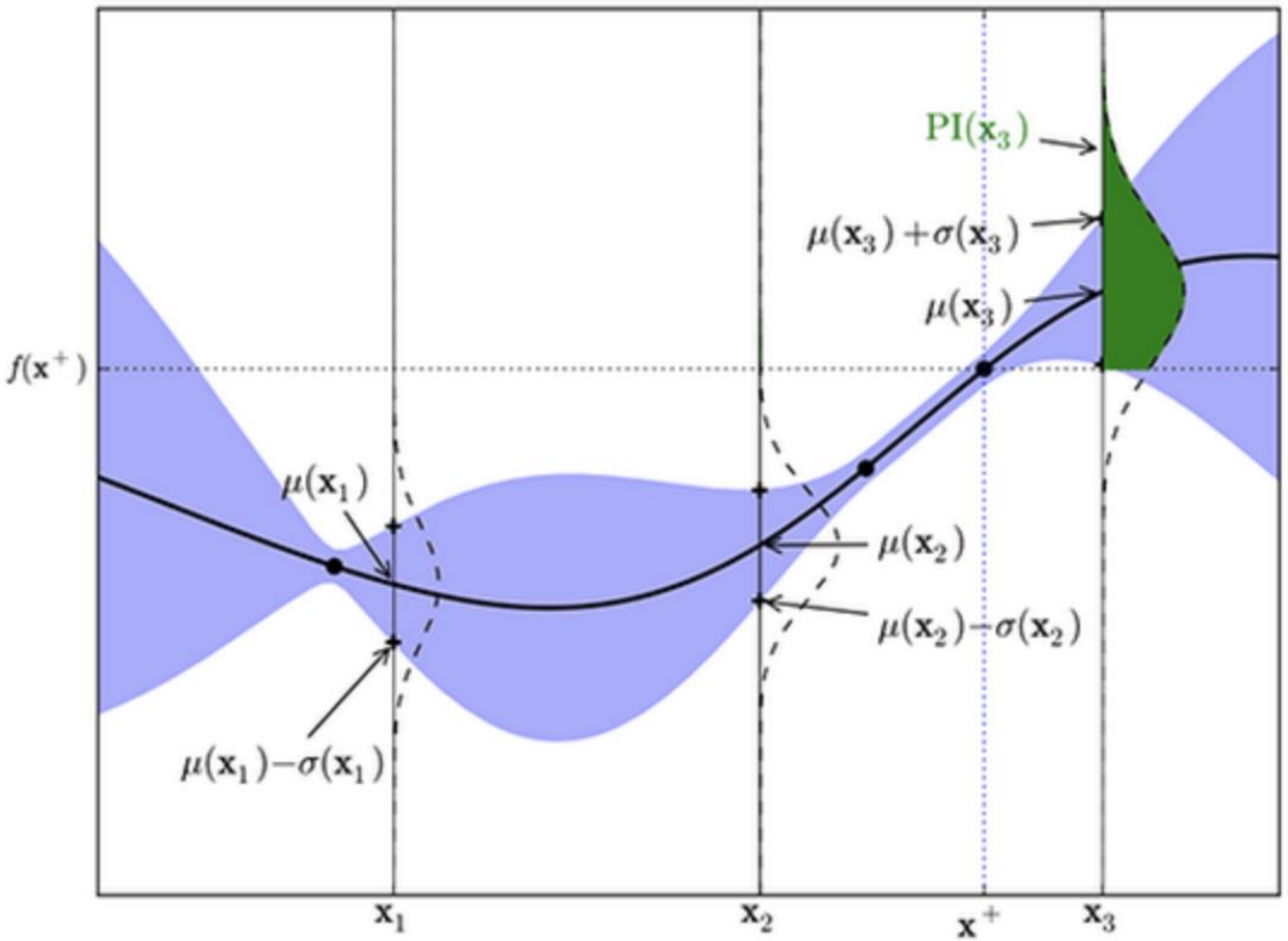
- Bayesian optimization
- Acquisition function

# Gaussian Process and Bayesian Inference




- We have some measurements in space X, and we want to maximize some property f(X).
- We create surrogate model: function and uncertainty over full parameter space based on measurements
- Gaussian Process: purely data driven
- Bayesian Inference: known model and some idea on parameters
- Structured Gaussian Process: physics-derived mean function

# Acquisition Functions (Policies)



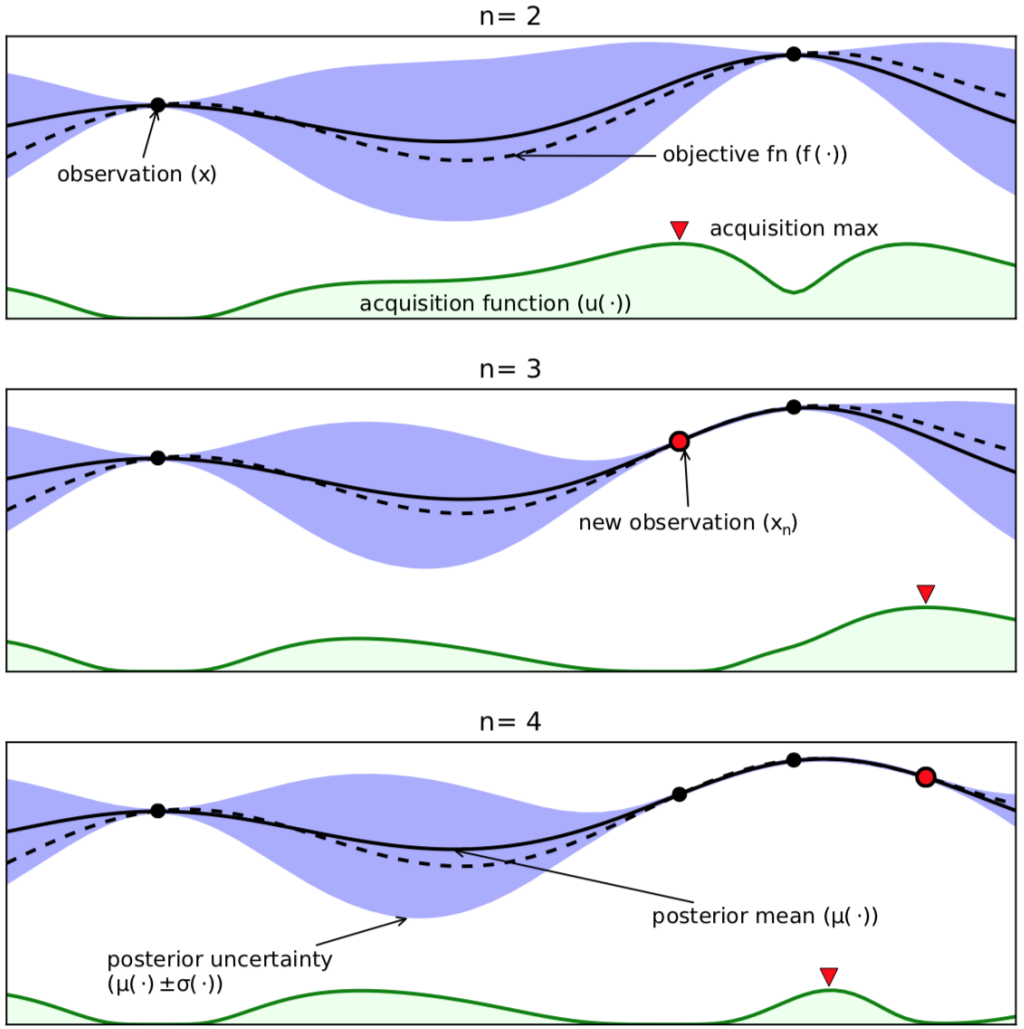

1. **Upper confidence bound:** simplest possible - just take the upper confidence bound from the prediction

2. **Probability of Improvement:** Integral from current functional maximum to upper limit of distribution as test point

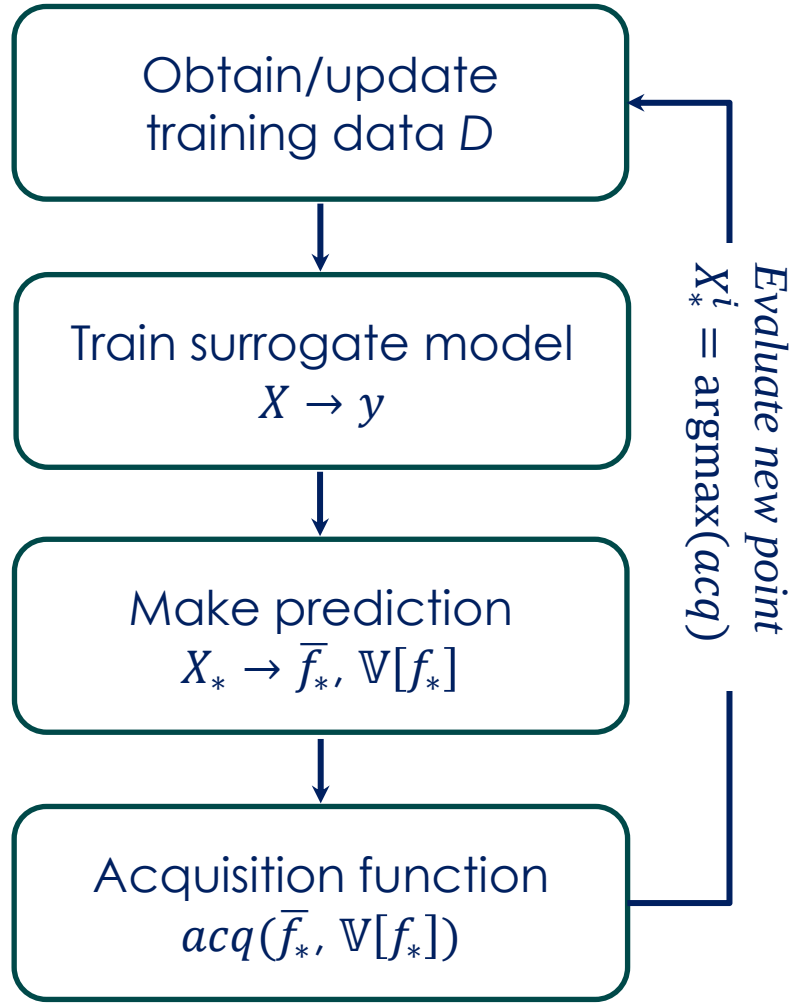
3. **Expected Improvement:** Instead of probability of improvement, we want to maximize the expected increase in the function value

4. **There are (always) more…**

# Bayesian Optimization



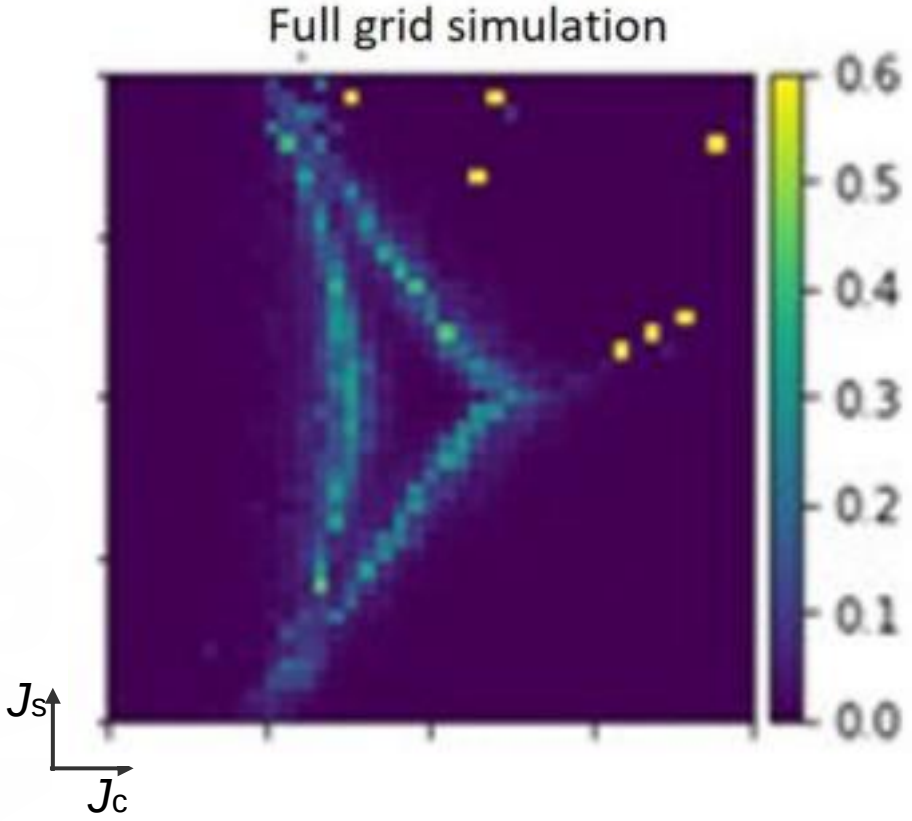

$X, y$: (sparse) Training data
$X_*$: New (not yet evaluated) points

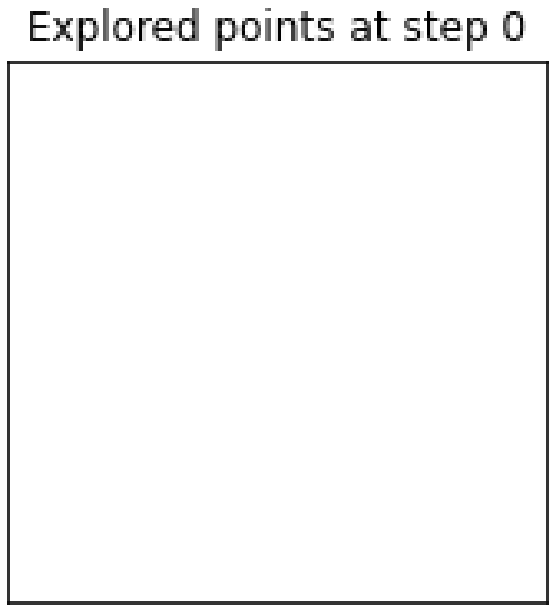


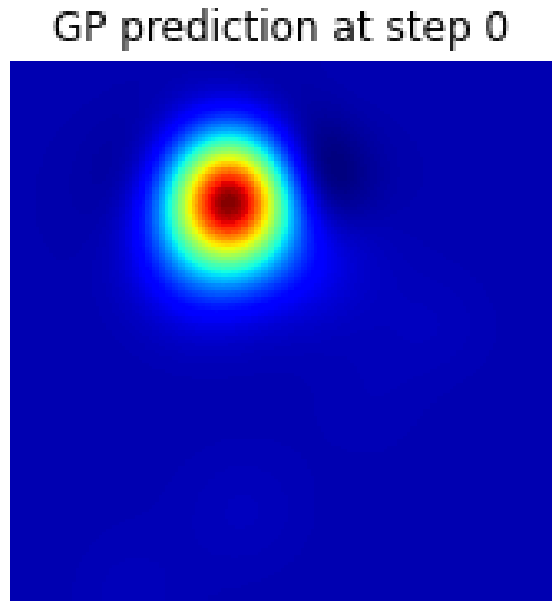

N. de Freitas et al., Taking the Human Out of the Loop: A Review of Bayesian Optimization , *Proceedings of the IEEE* **104**, 148 (2015)

# Bayesian Optimization for Physical Discovery

**Discovering regions in which the heat capacity is maximized in NNN Ising model**



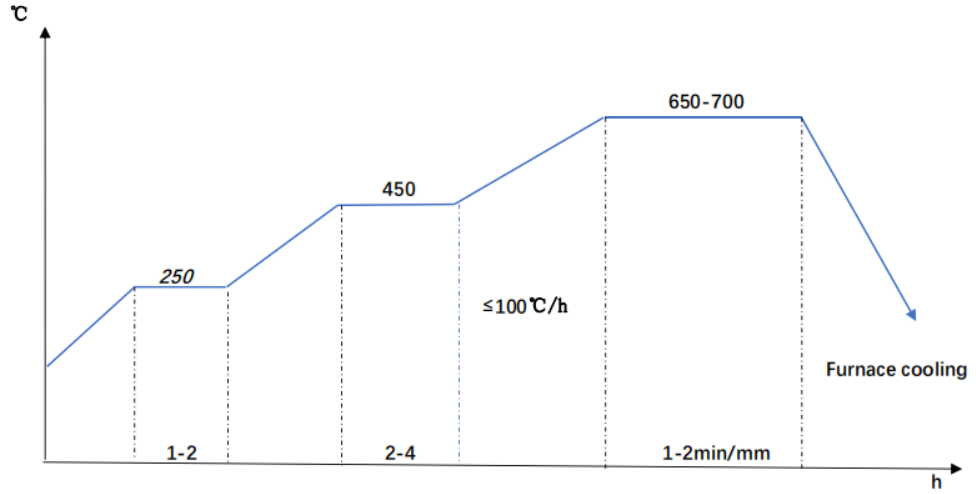
# What is the limitation of the GP/BO?

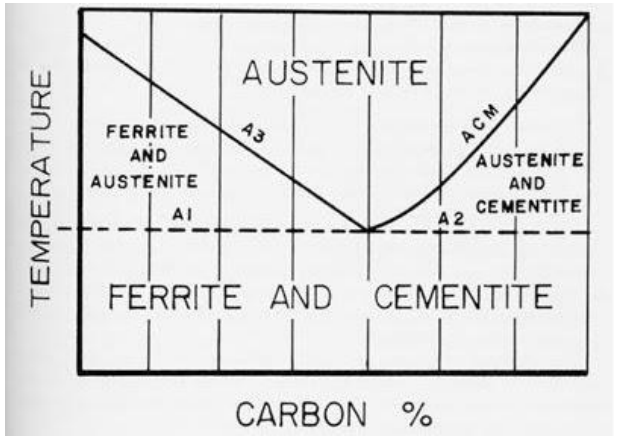
1. Works only in low-dimensional spaces
2. The correlations are defined by the kernel function (very limiting)
3. We do not use any knowledge about physics of the system
4. We do not use cheap information available during the experiment (proxies)

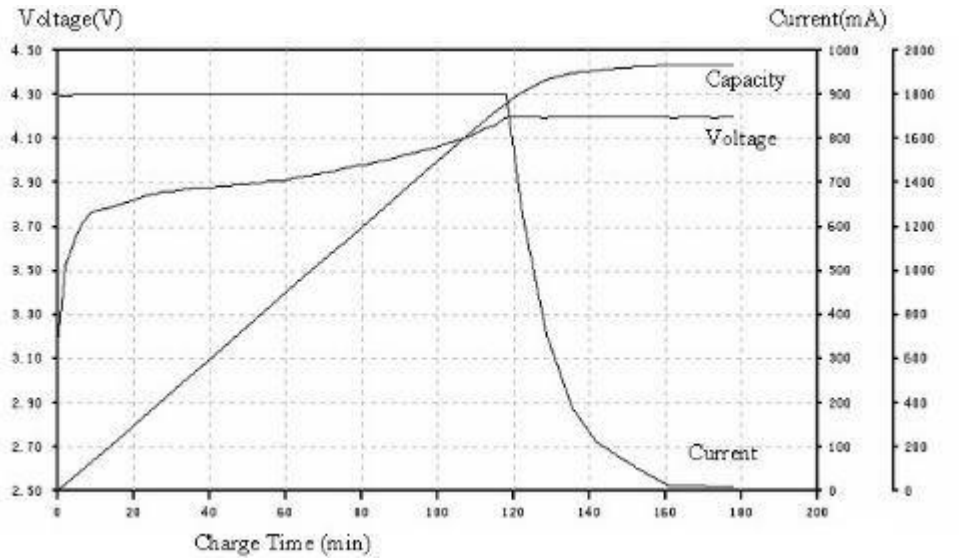
# Can we somehow make high dimensional space low-D?

1. Structure-property relationships
2. Molecular discovery and QASR
3. Processing optimization

# Making materials: process trajectories



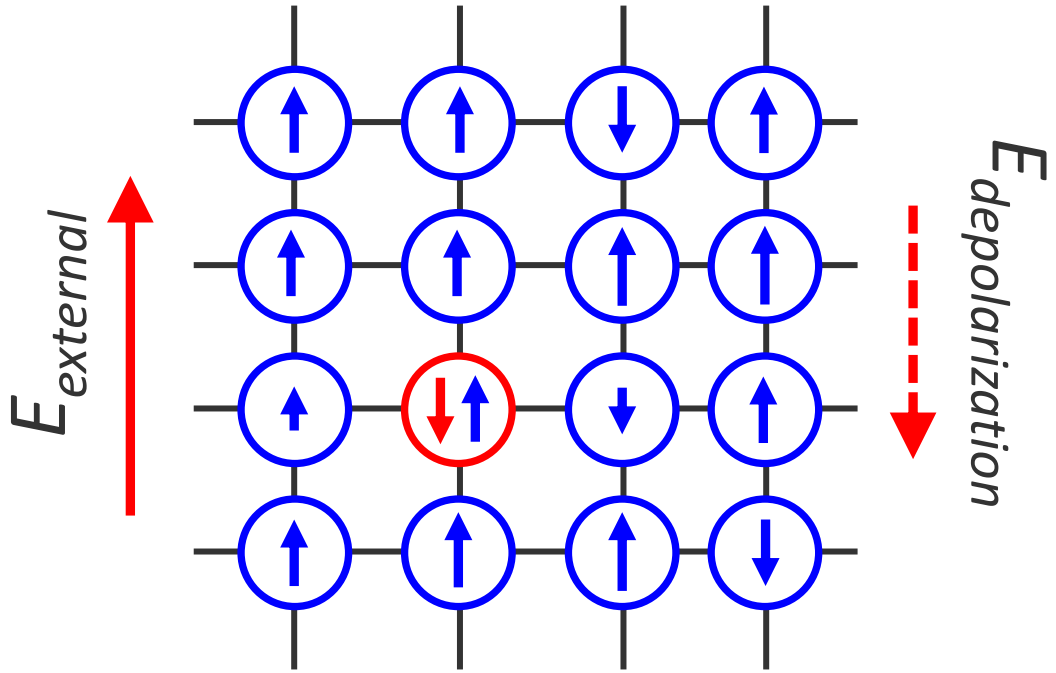



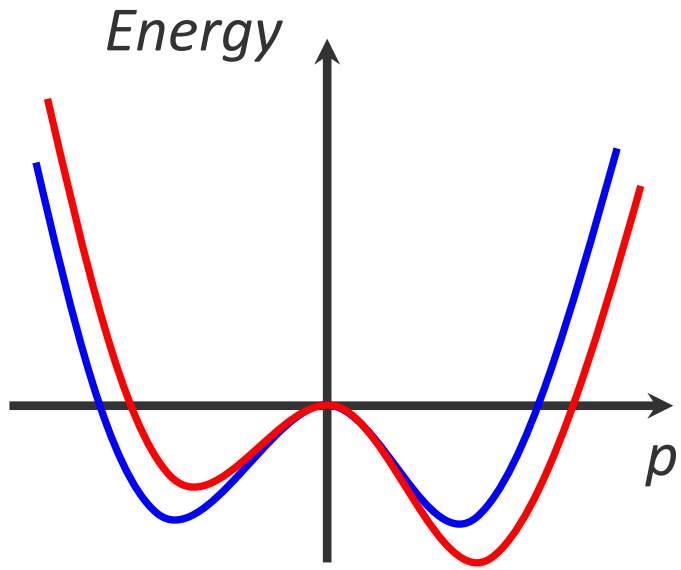




- Making steel: complicated and took a lot of time optimize
- Charging battery: obvious economic impact
- Manufacturing: Annealing hybrid perovskite thin films
- Poling ferroelectric

How do we optimize trajectories if we have (a) only limited or no mechanistic information, (b) our experimental budgets are limited, but (c) we have some access to domain expertise?

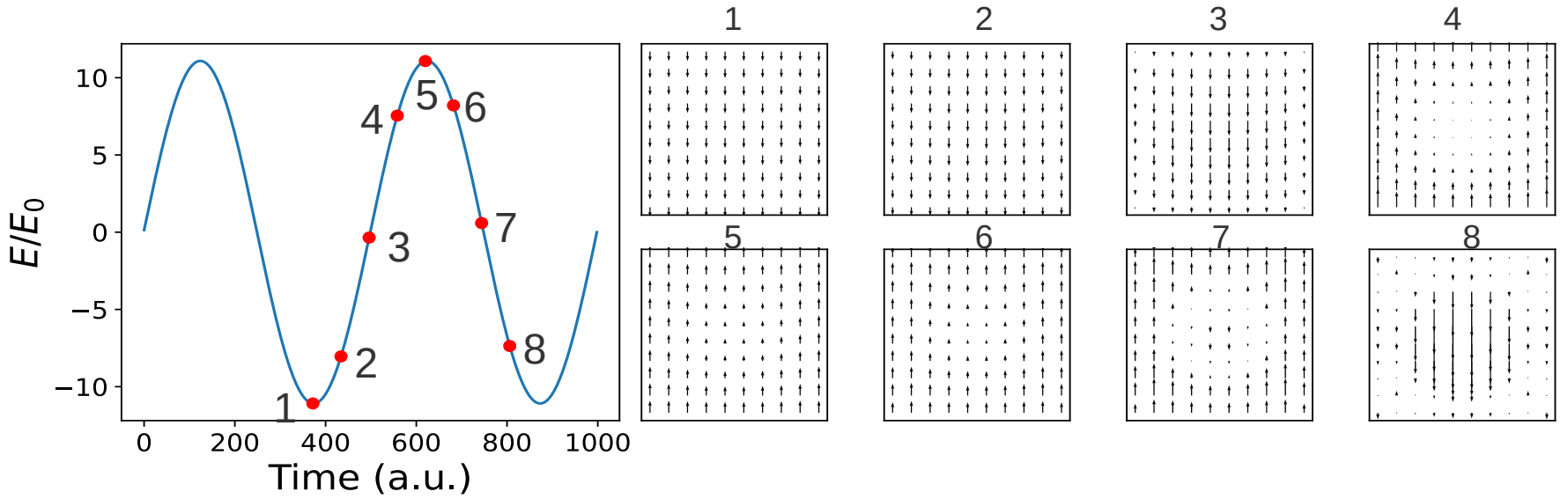
# FerroSIM: the simplest interesting ferroelectric

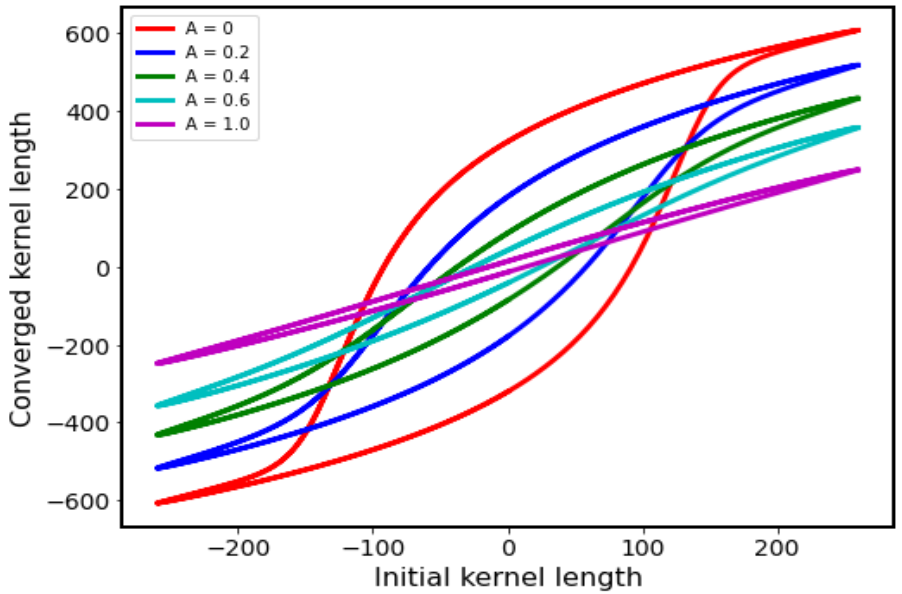


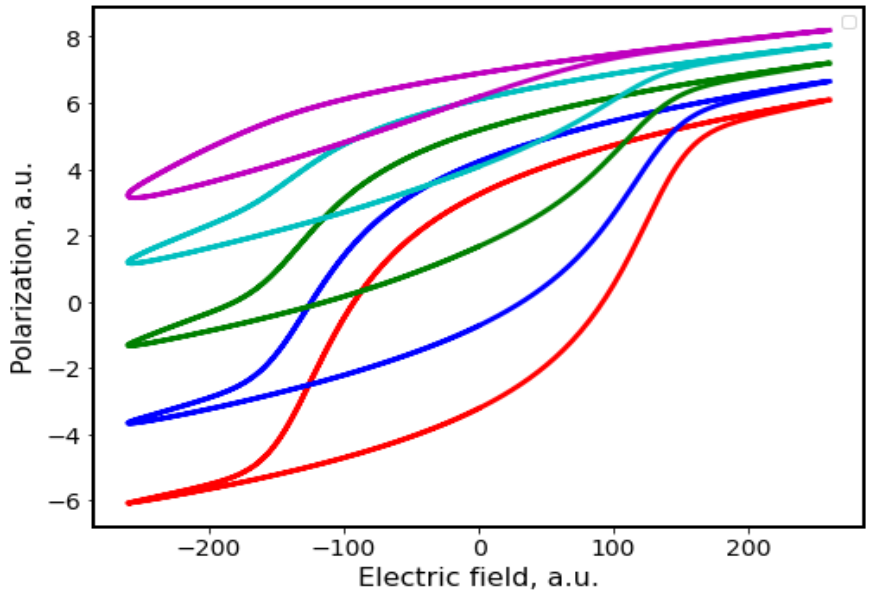




- A discrete square lattice where a continuous polarization vector resides at each lattice site
- The local free energy at each site takes the GLD form:
  - $F_{ij} = \alpha_1 \left(p_{x_{ij}}^2 + p_{y_{ij}}^2\right) + \alpha_2 \left(p_{x_{ij}}^4 + p_{y_{ij}}^4\right) + \alpha_3 p_{x_{ij}}^2 p_{y_{ij}}^2 - E_{loc_{x_{ij}}} p_{x_{ij}} - E_{loc_{y_{ij}}} p_{y_{ij}}$
  - Where, $E_{loc} = E_{ext} + E_{dep} + E_d(i,j)$ and $E_d = -\alpha_{dep} < p >$
- The total free energy is the sum of local free energies and coupling terms:
  - $F = \sum_{i,j}^{N} F_{ij} + K \sum_{k,l} (p_{x_{ij}} - p_{x_{i+k,j+l}})^2 + K \sum_{k,l} (p_{y_{ij}} - p_{y_{i+k,j+l}})^2$
- Polarization at each lattice site is updated to decrease the free energy using $\frac{\gamma dp_{i,j}}{dt} = -\frac{\partial F}{\partial p_{i,j}}$

# Microstates and Macroscopic Observables

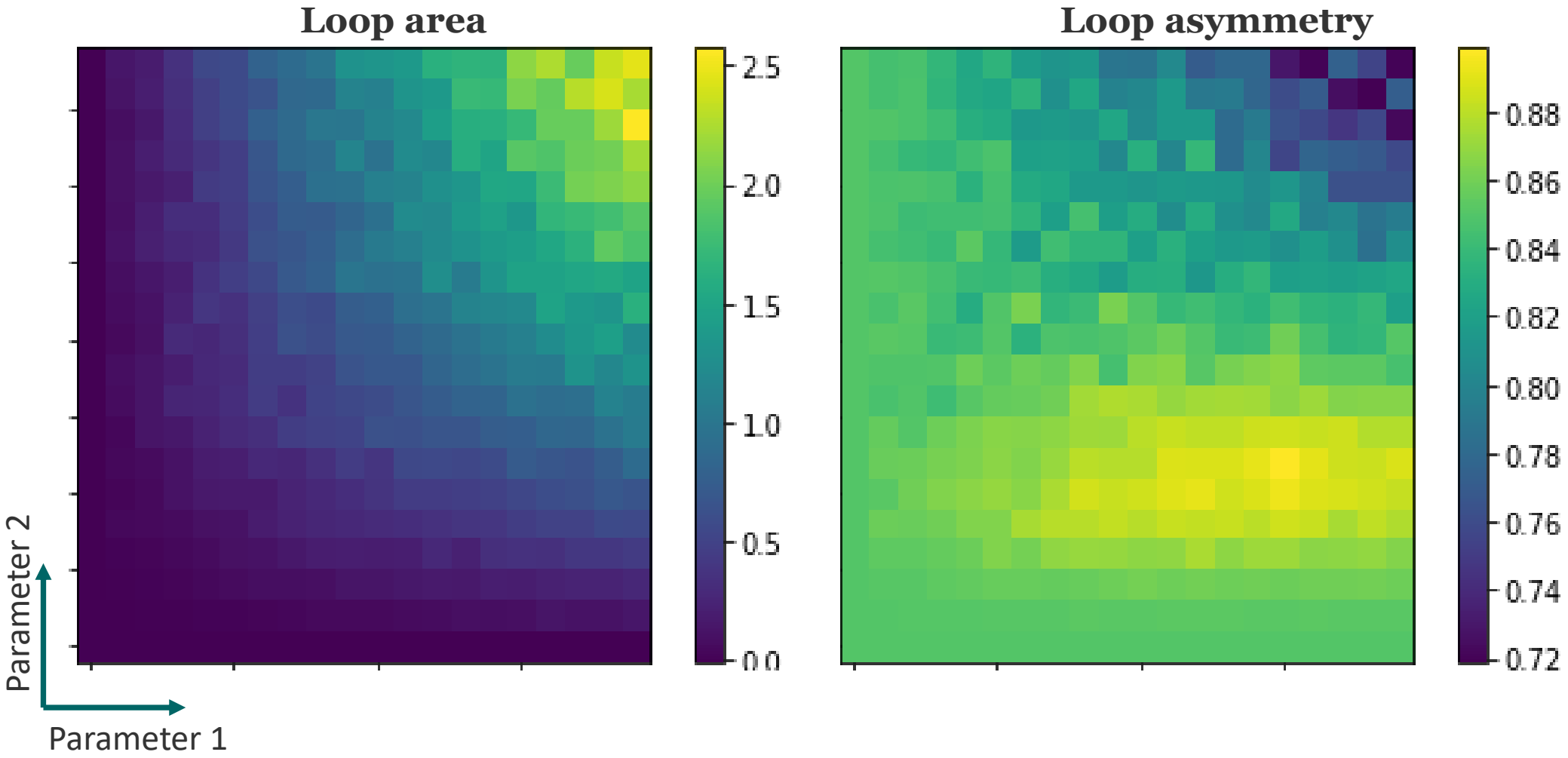


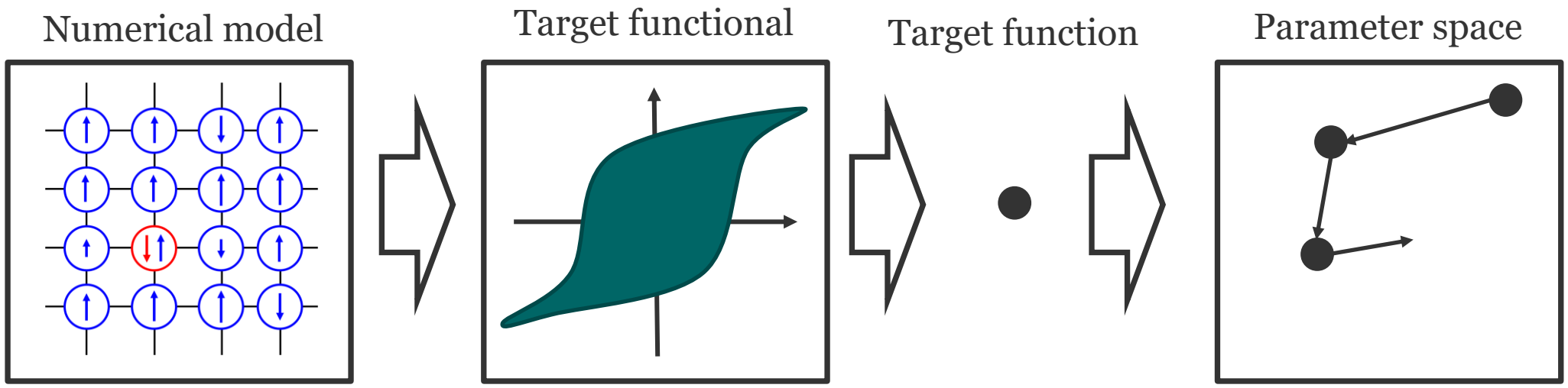
# Global response vs. model parameters

Parameter space 1: Hamiltonian
Parameter space 2: Field history



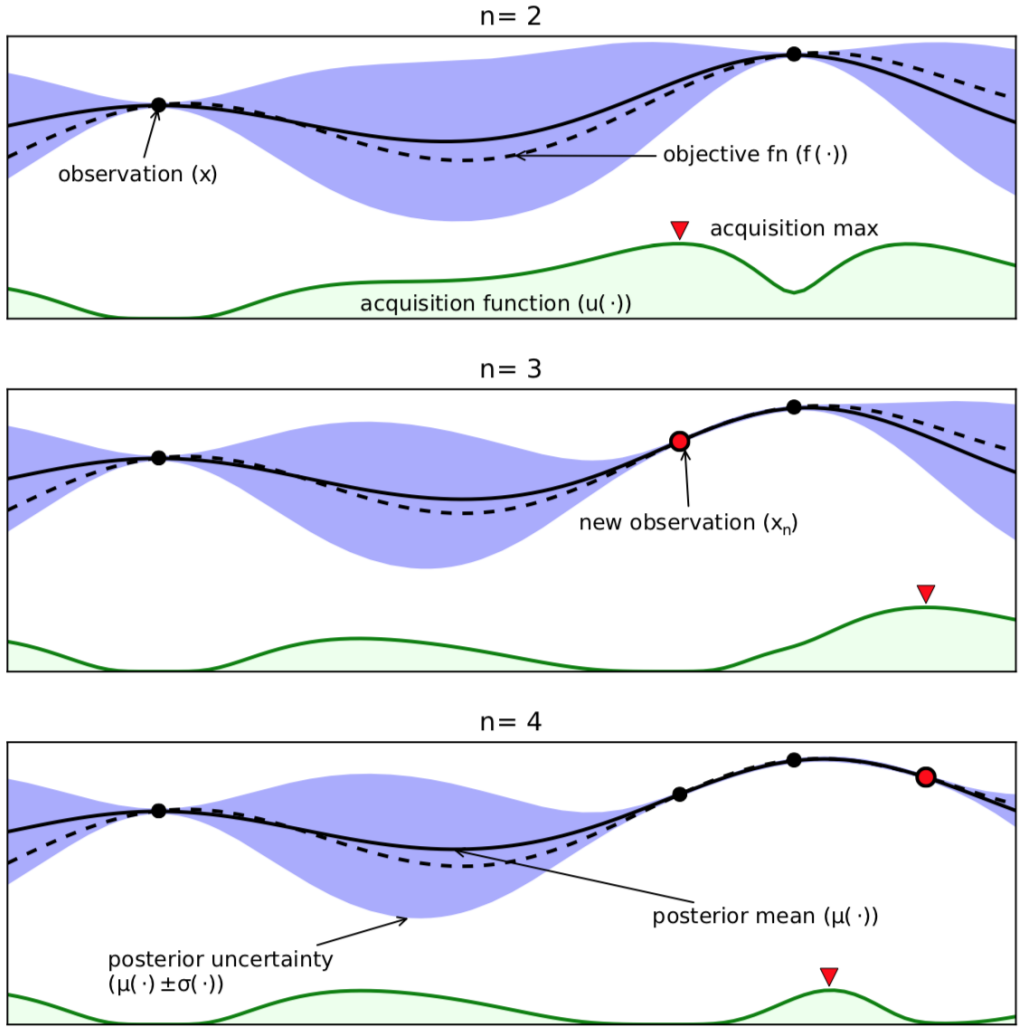

For small dimensional parameter spaces, we can evaluate global responses via the grid search

# Can we do better then grid search?



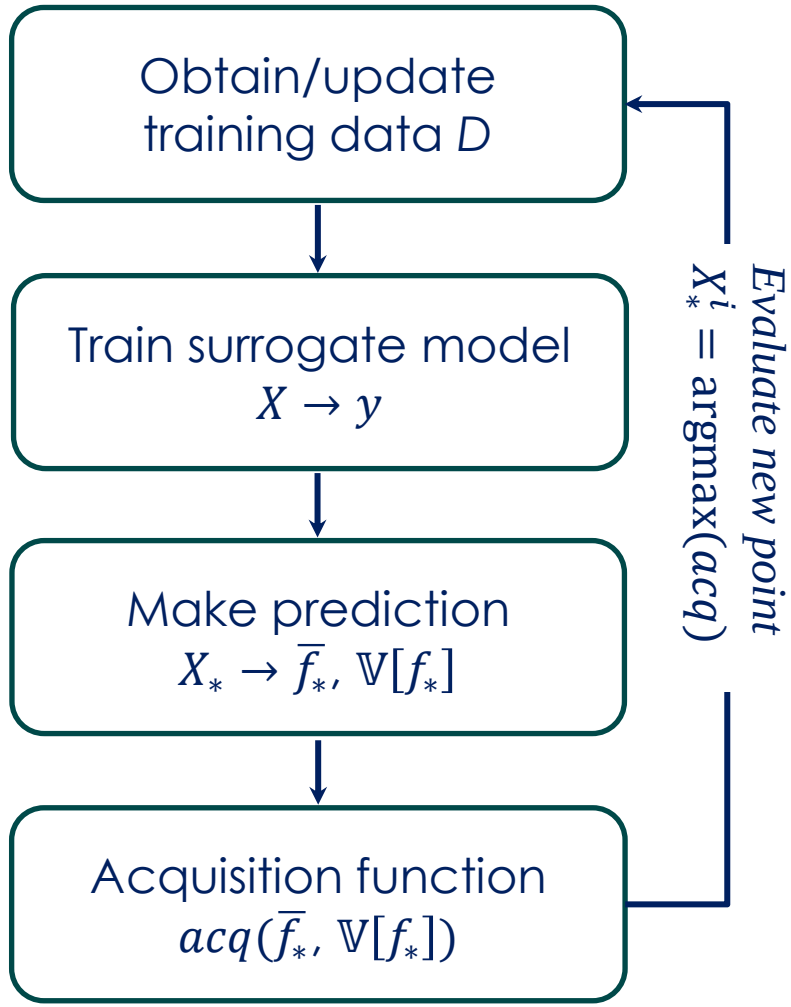

Or we can use simple Gaussian Process-based Bayesian Optimization to do so

# Bayesian Optimization!

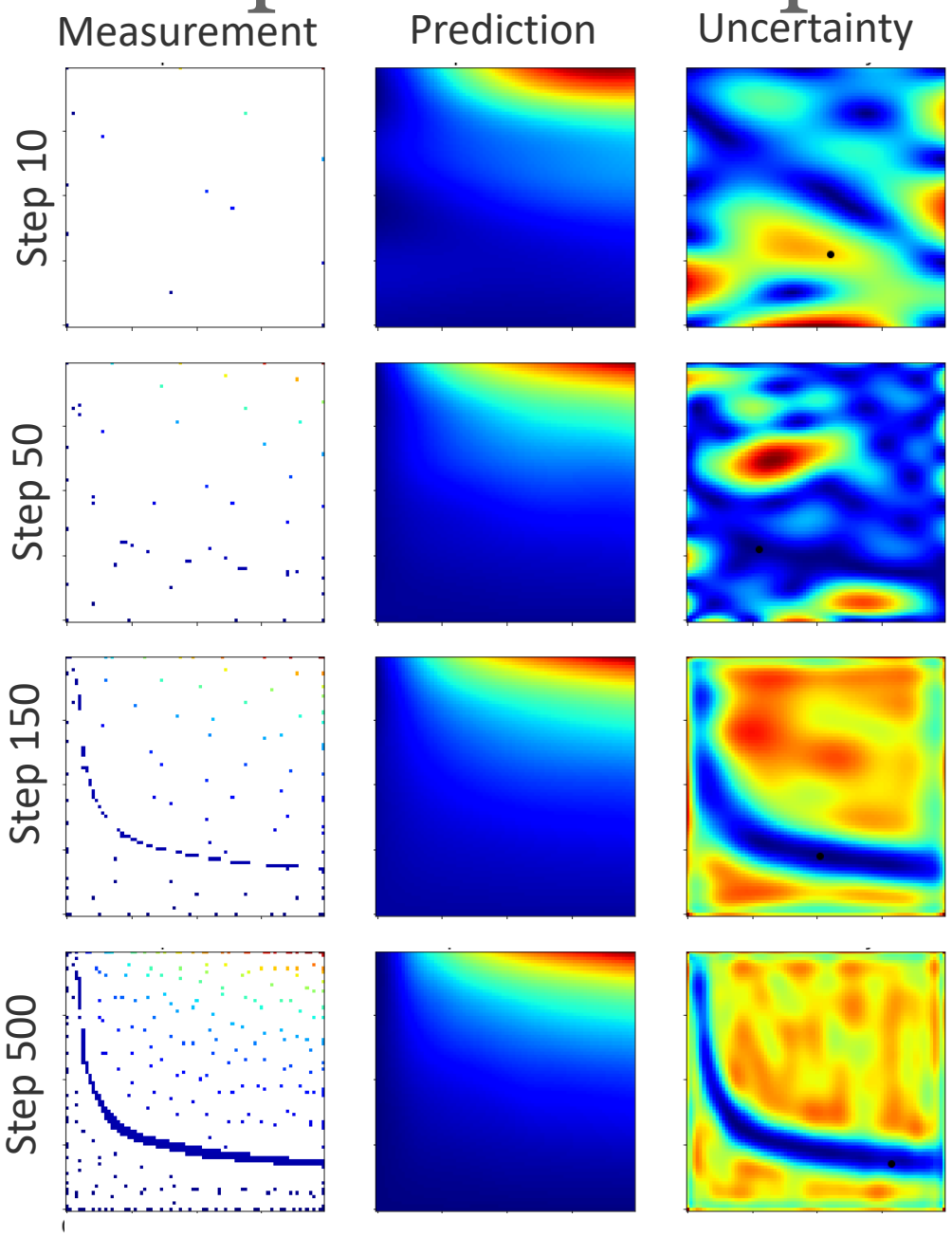



N. de Freitas et al., Taking the Human Out of the Loop: A Review of Bayesian Optimization , *Proceedings of the IEEE* **104**, 148 (2015)

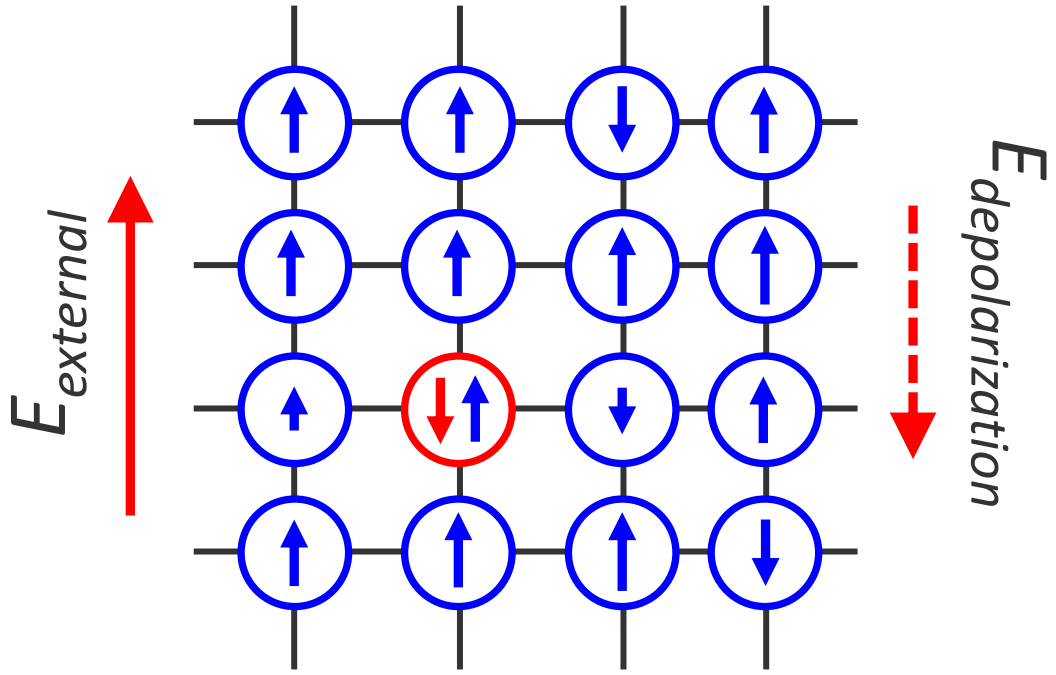
$X, y$: (sparse) Training data
$X_*$: New (not yet evaluated) points

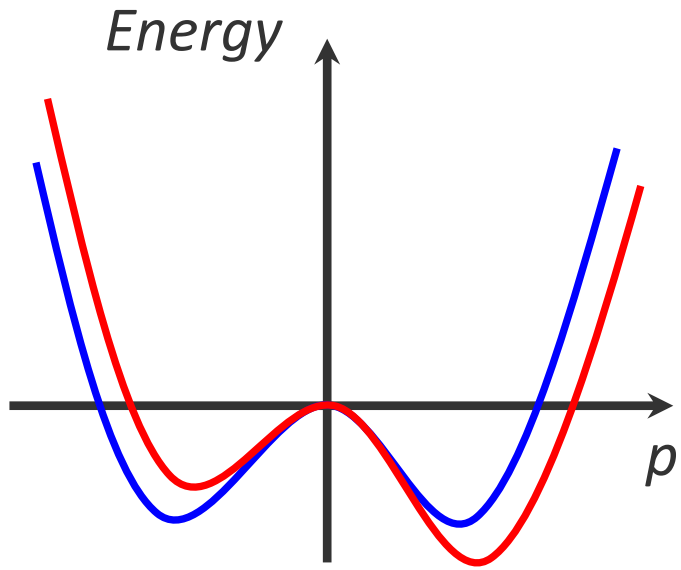


# BO exploration of parameter space



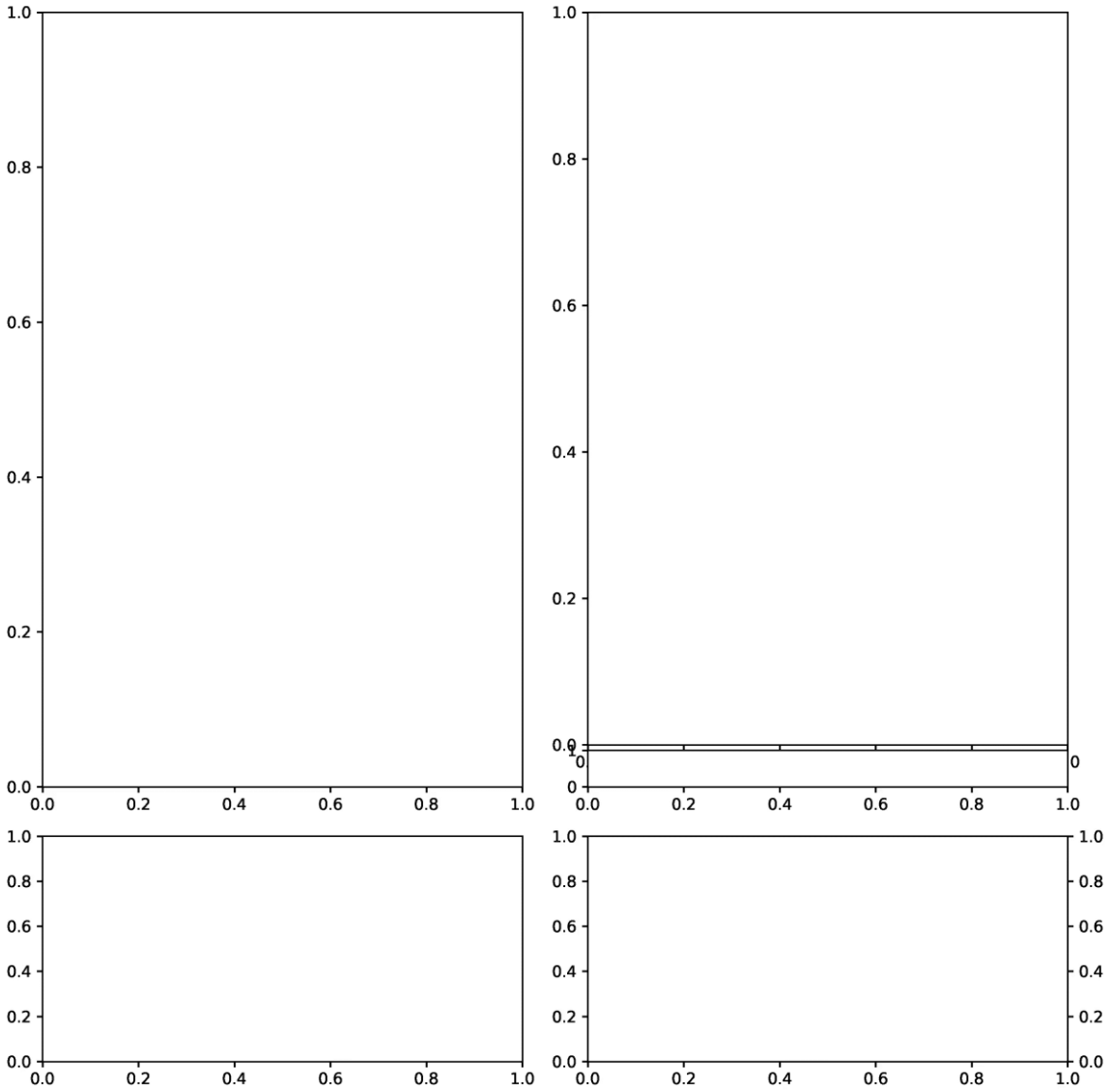
# FerroSIM: the simplest interesting ferroelectric




- A discrete square lattice where a continuous polarization vector resides at each lattice site
- The local free energy at each site takes the GLD form:
  - $F_{ij} = \alpha_1 \left(p_{x_{ij}}^2 + p_{y_{ij}}^2\right) + \alpha_2 \left(p_{x_{ij}}^4 + p_{y_{ij}}^4\right) + \alpha_3 p_{x_{ij}}^2 p_{y_{ij}}^2 - E_{loc_{x_{ij}}} p_{x_{ij}} - E_{loc_{y_{ij}}} p_{y_{ij}}$
  - Where, $E_{loc} = E_{ext} + E_{dep} + E_d(i,j)$ and $E_d = -\alpha_{dep} < p >$
- The total free energy is the sum of local free energies and coupling terms:
  - $F = \sum_{i,j}^{N} F_{ij} + K \sum_{k,l} (p_{x_{ij}} - p_{x_{i+k,j+l}})^2 + K \sum_{k,l} (p_{y_{ij}} - p_{y_{i+k,j+l}})^2$
- Polarization at each lattice site is updated to decrease the free energy using $\frac{\gamma dp_{i,j}}{dt} = -\frac{\partial F}{\partial p_{i,j}}$

# But what about trajectories?



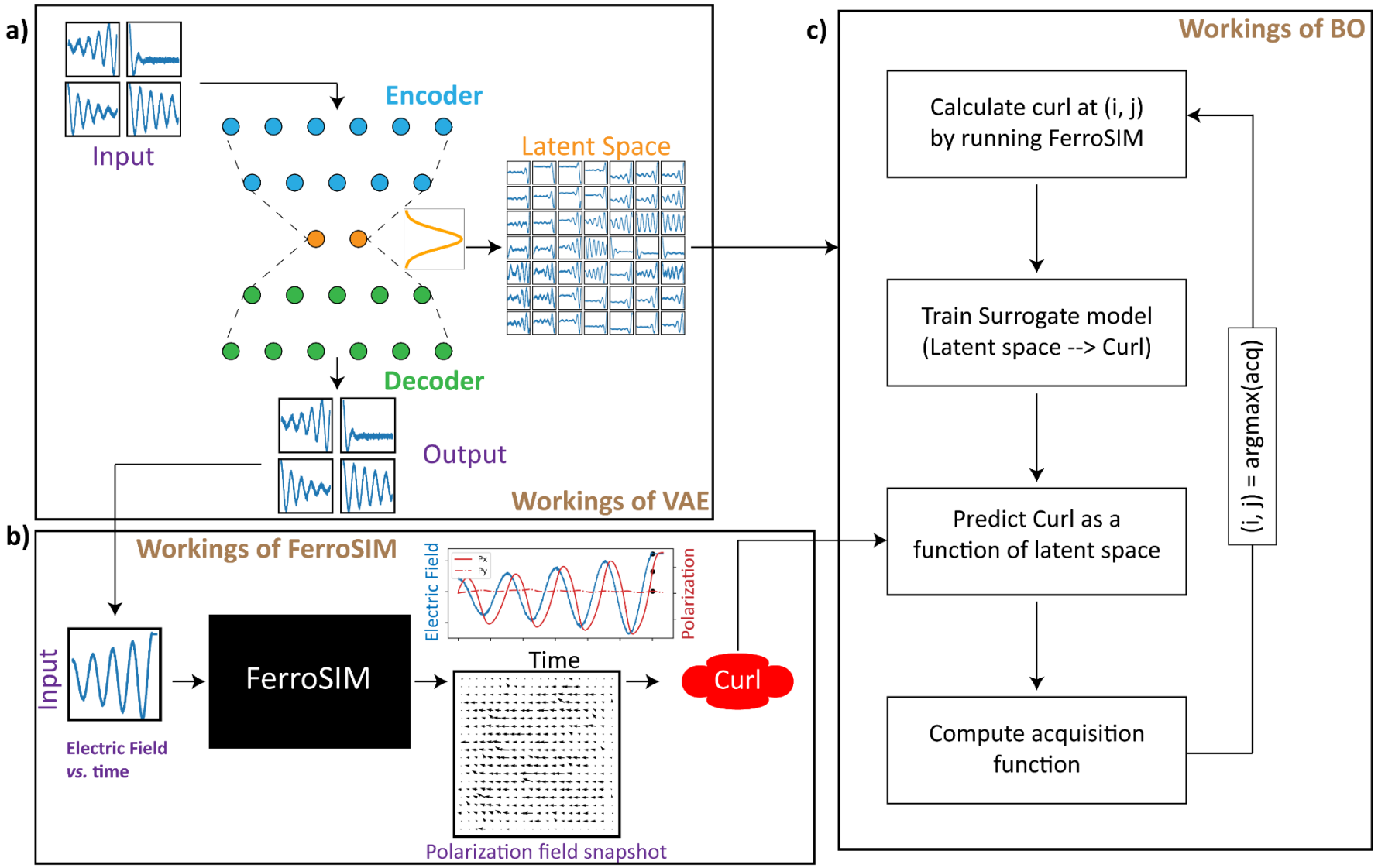
- The model has large number of microstates
- The global state depends on history, i.e. dependence of field vs. time
- Can we somehow optimize the chosen global state in the space of possible histories?

- This space is obviously intractable…

- … however, we are not interested in ALL possible histories. We are interested in relatively simple histories

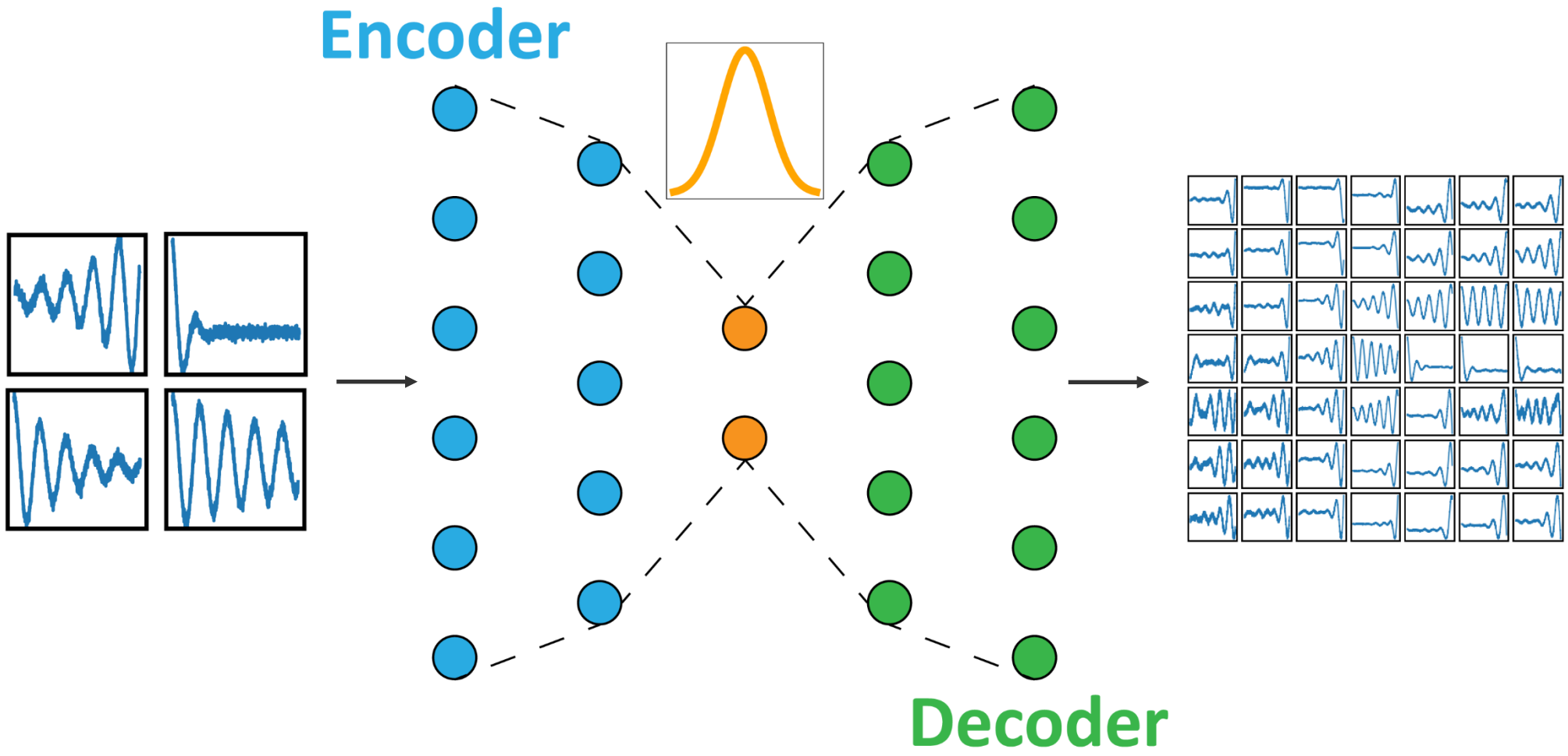
- **Thought:** what if we start with the histories that make sense from domain perspective, and look  for way to simplify them?

# Putting everything together




Same approaches are used for molecular discovery, polymer, and biomolecules

# Can VAE help?

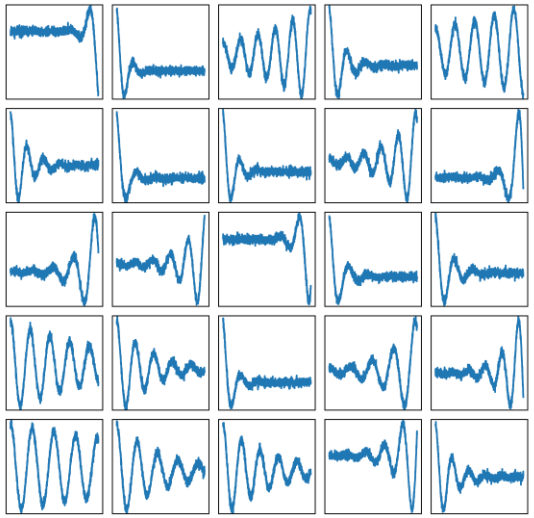


# VAE encoding of domain trajectories

- Sinusoidal trajectories with exponential functions as amplitude modulators
  - $Aexp(\alpha t)\sin(\omega t) + B$
- $A$: [0, 0.75],
- $\alpha$: [-2.75, 2.75],
- $\omega$: [$2\pi$, $\frac{8}{3}\pi$],
- $B$: [-0.5, 0.5]

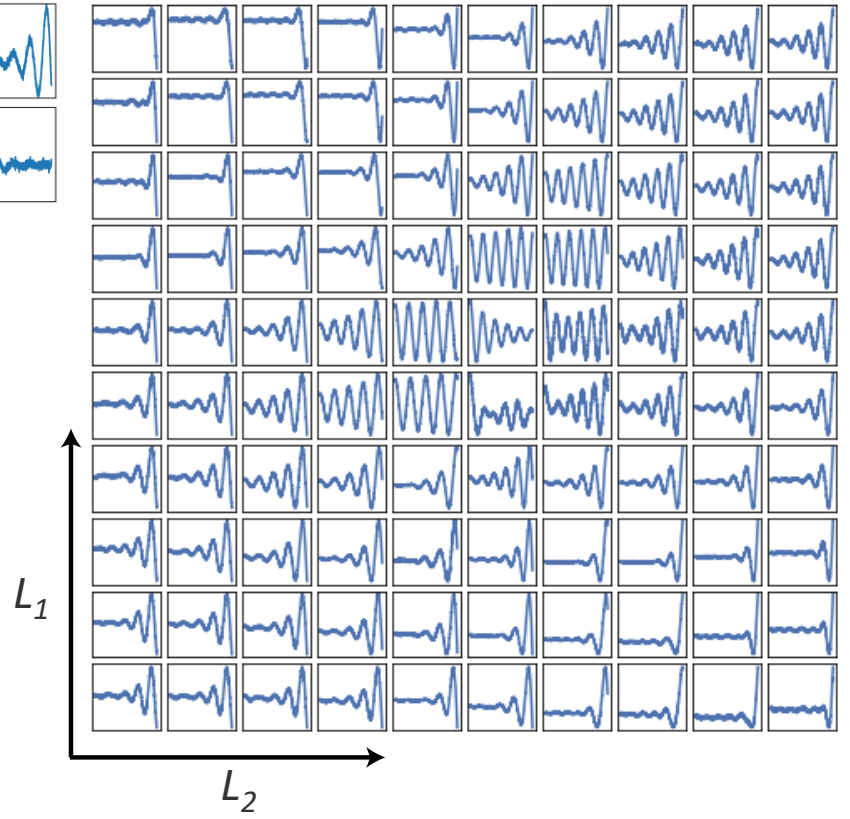
- These electric fields are divided into 900 discrete time steps.

- 7500 of these curves are then used to create a smooth latent space using a Variational Autoencoder (VAE)

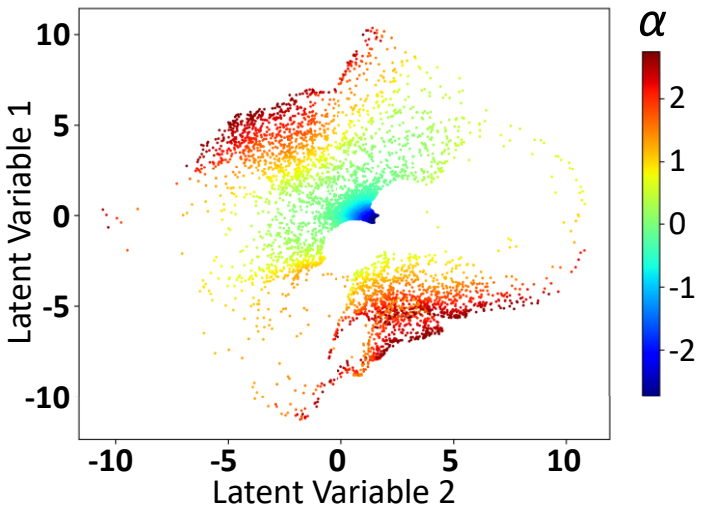


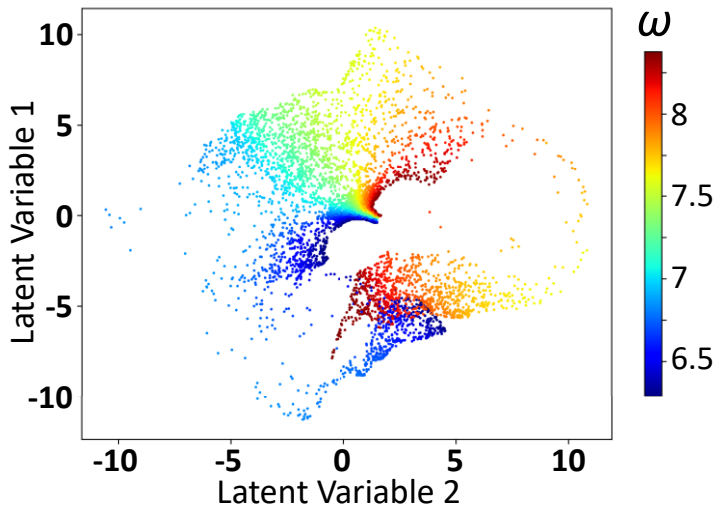
Example Trajectories

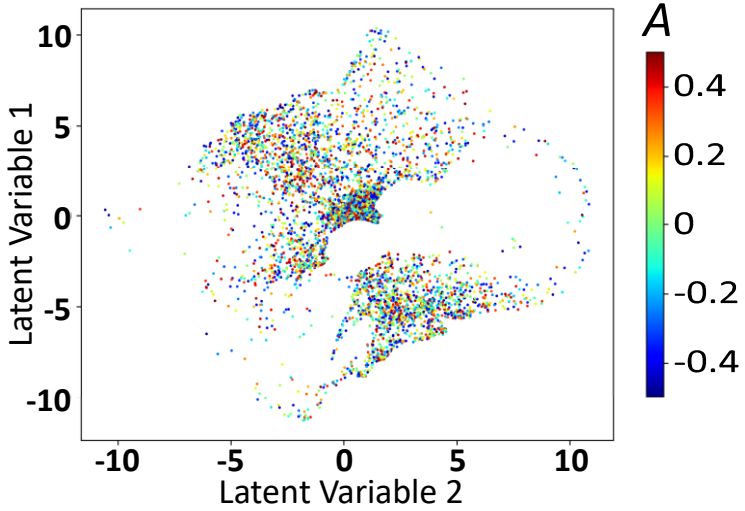
Decoded Latent Space



# Latent space distributions

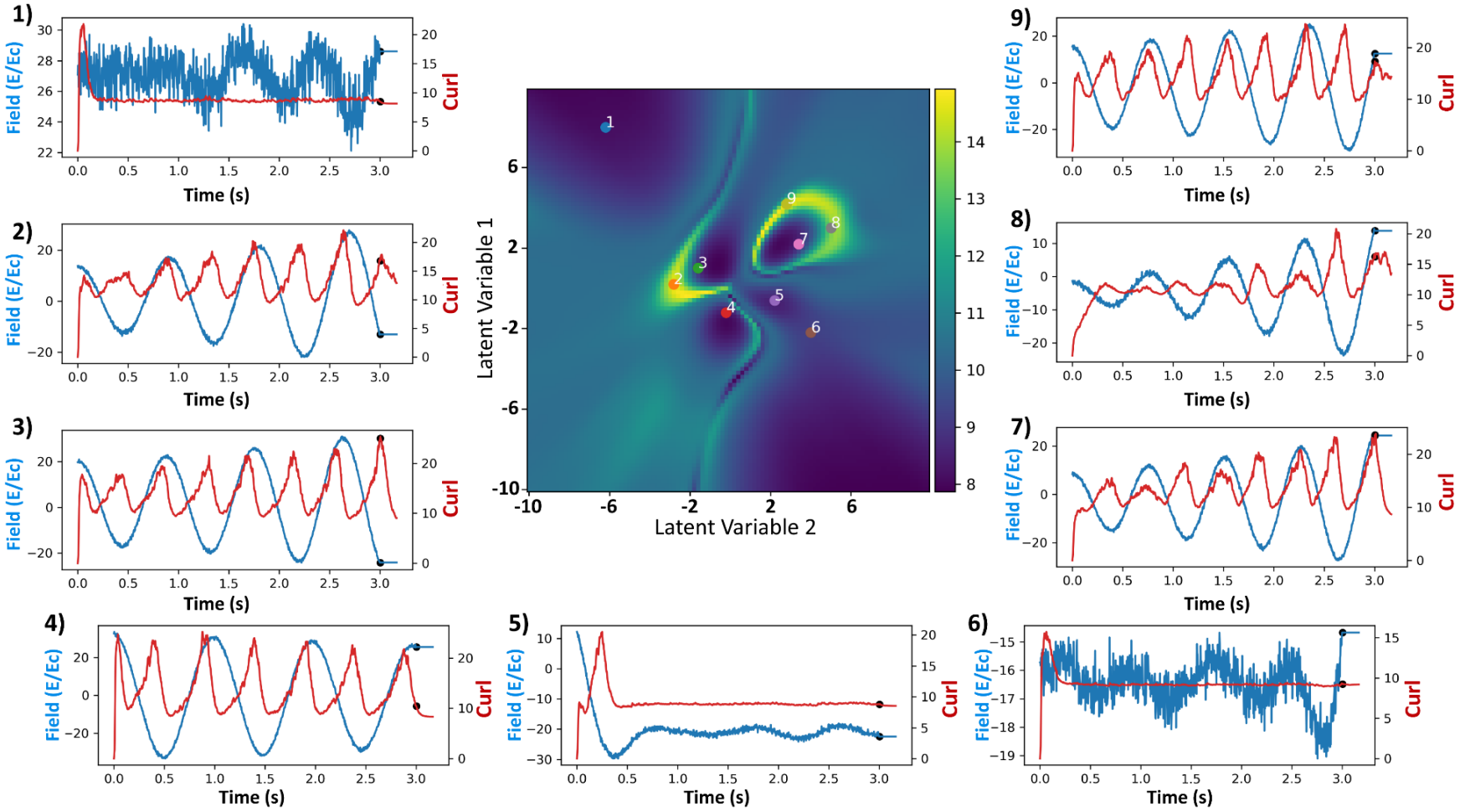


# Ground truth target function

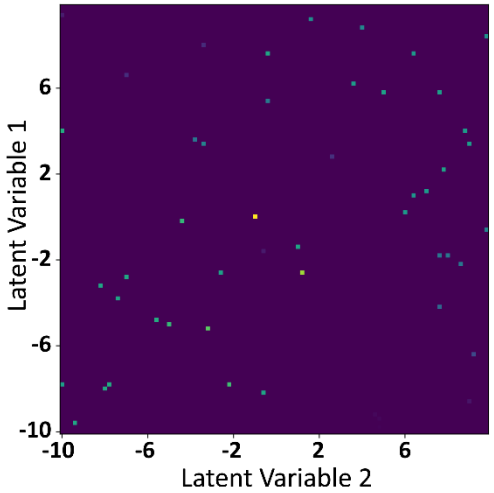
- Latent space is sampled and then decoded back into the space of electric field of 900-dimensions
- An equilibration region of 50-time steps is then added where the electric field is held constant at the final value of the decoded electric field.
- The sum of absolute value of curl at each lattice site at the end of the simulation is the target value to be optimized



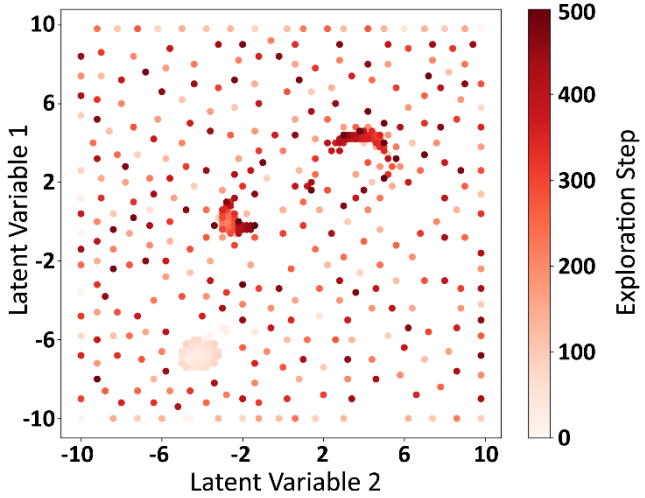

- Curl decays in the equilibration region
- The rate of decay of the curl is proportional to the curl at the onset of the equilibration region
- The local maxima of the curl seemingly coincides with the local optima of the electric field.

# Bayesian Optimization in the Latent Space

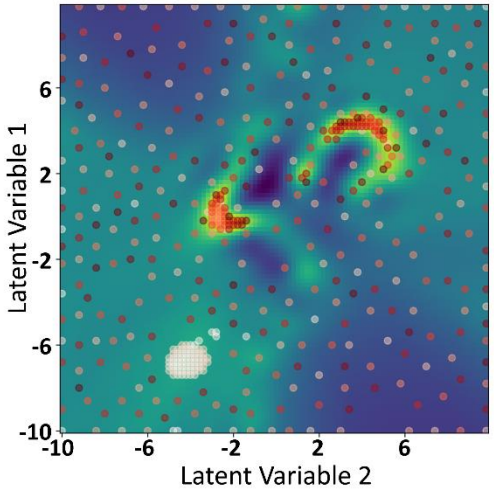
100 initial points



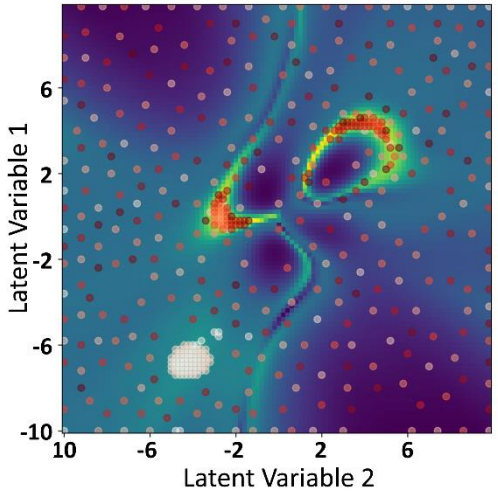

Explored points




Reconstructed curl surface

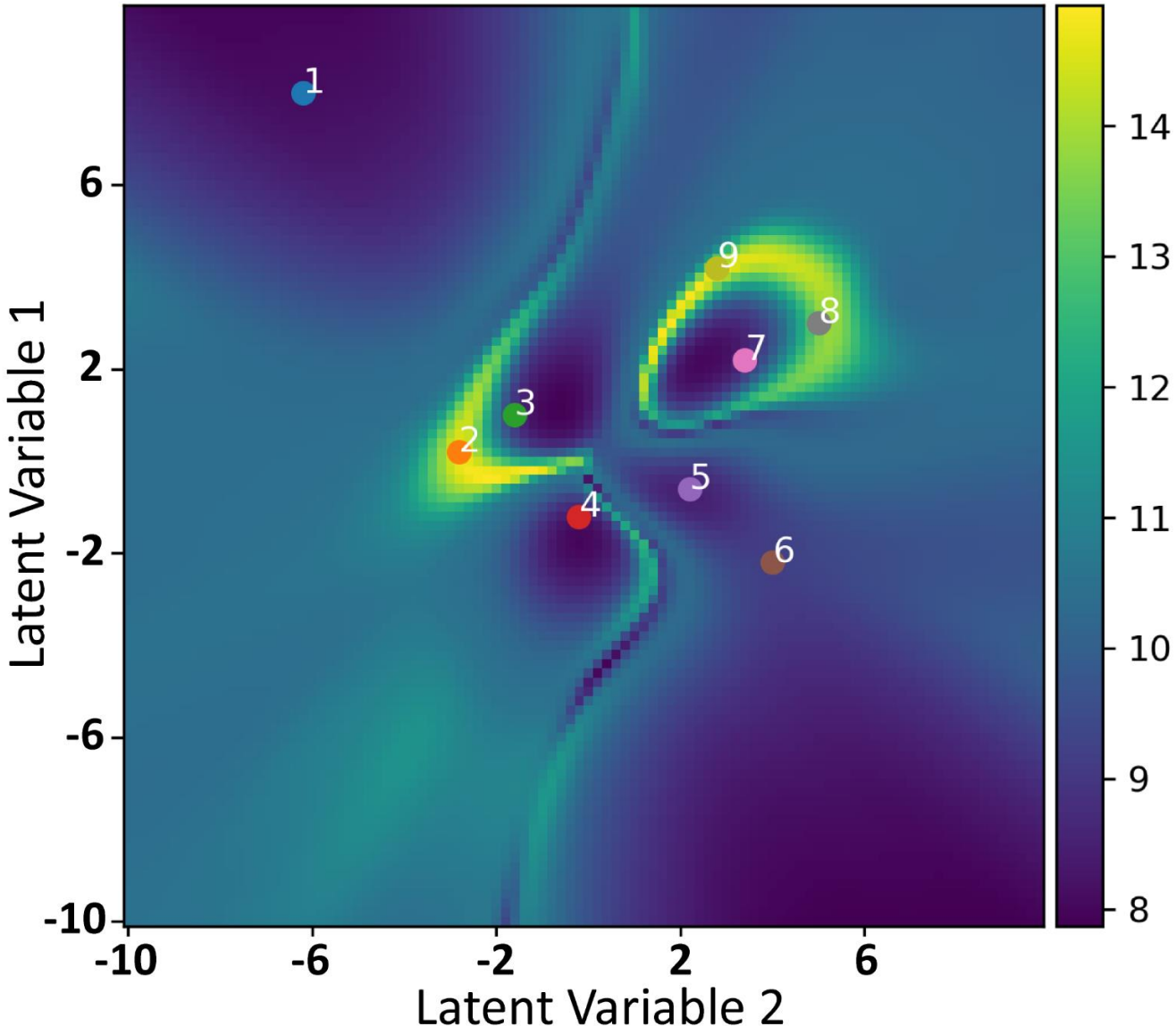



Original curl surface




- 100 initialization points and the BO explored the latent space for the next 500 points
- Acq function: $\mu + 10\sigma$
- So, at the end BO only explored a total of 600 points out of 10,000 points the latent space is divided into
- Caveat: we had to tune the Acq with the ground truth data known

# What determines success?

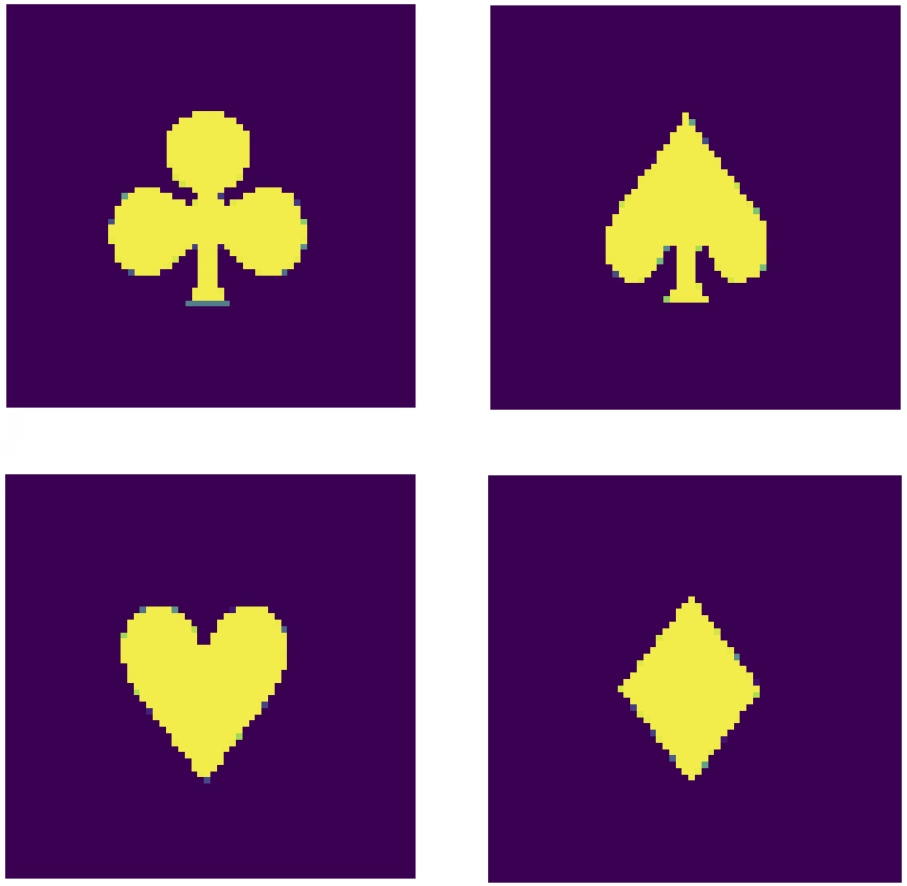



The success of the BO in the latent space clearly depends on the shape on the manifold that points of interest form.

For VAE, the shape of the manifold is determined by the properties of the data only, including

(a) how strong correlations in data reflect in correlation in properties and
(b) weight of the “good” trajectories

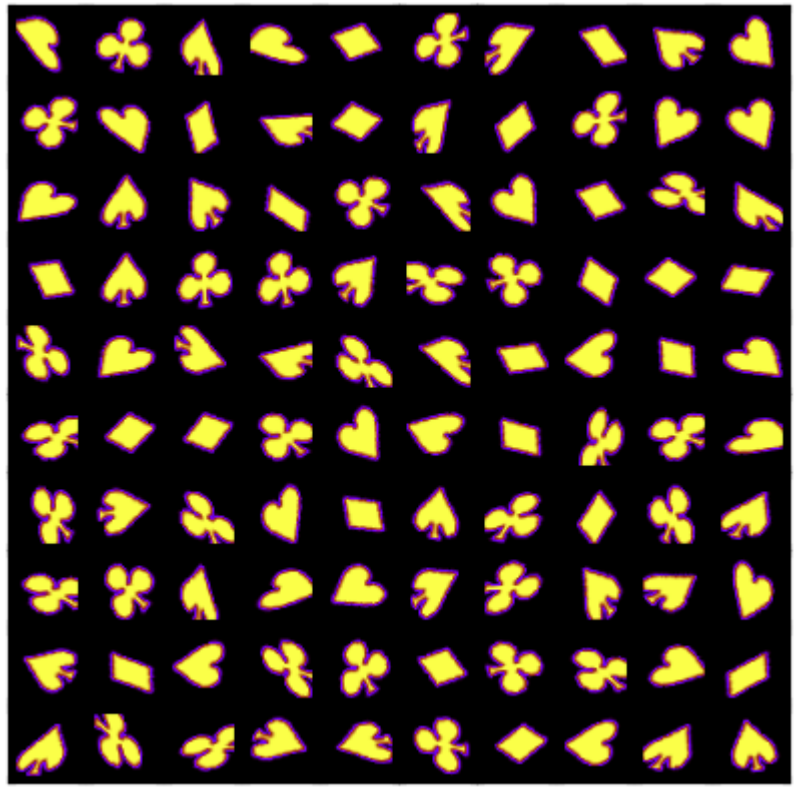
# Card data set



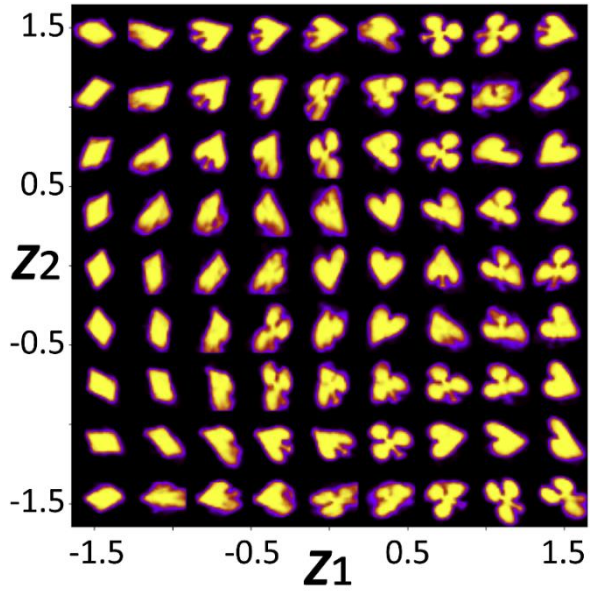
Rotations:
[-120˚, 120˚

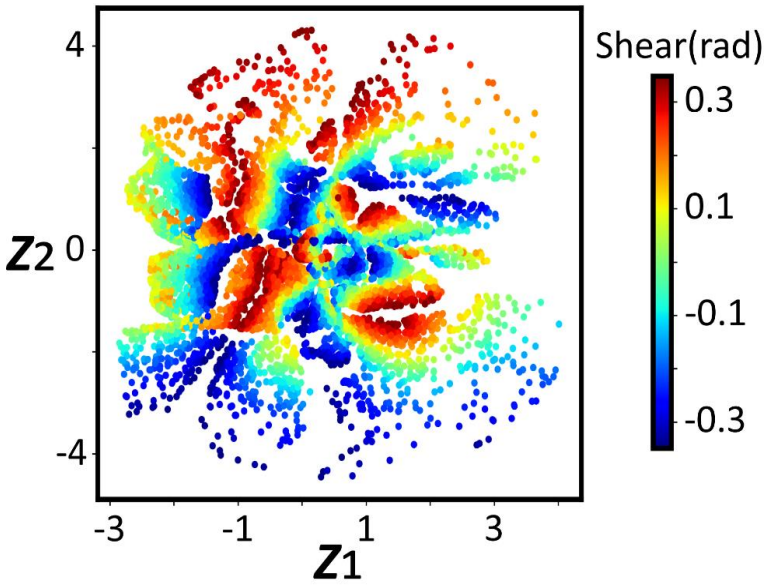
Shear:
[-20˚, 20˚ ]

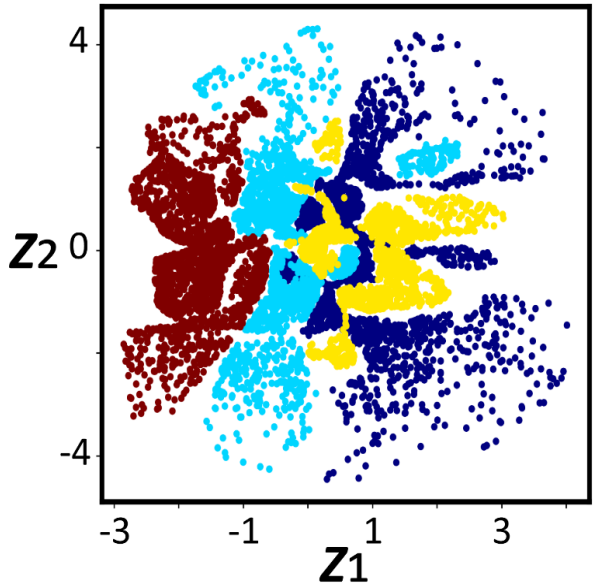
Randomly picked samples

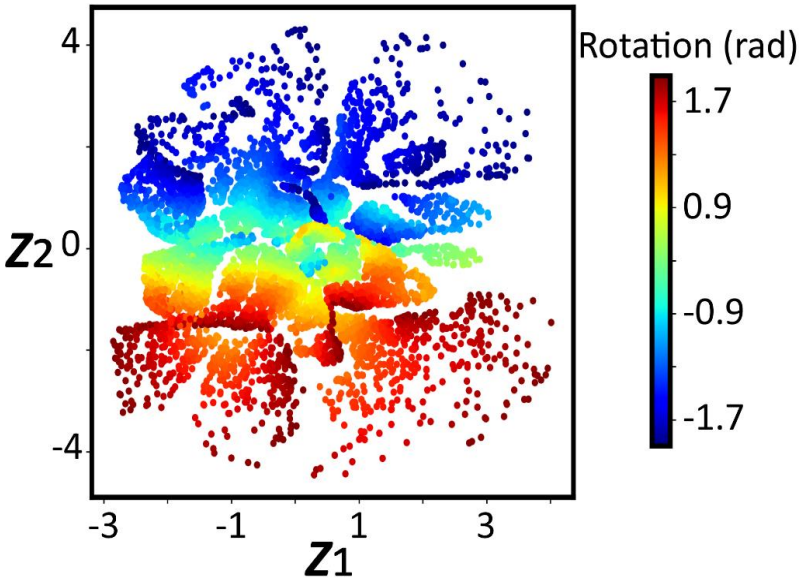


# VAE on Cards



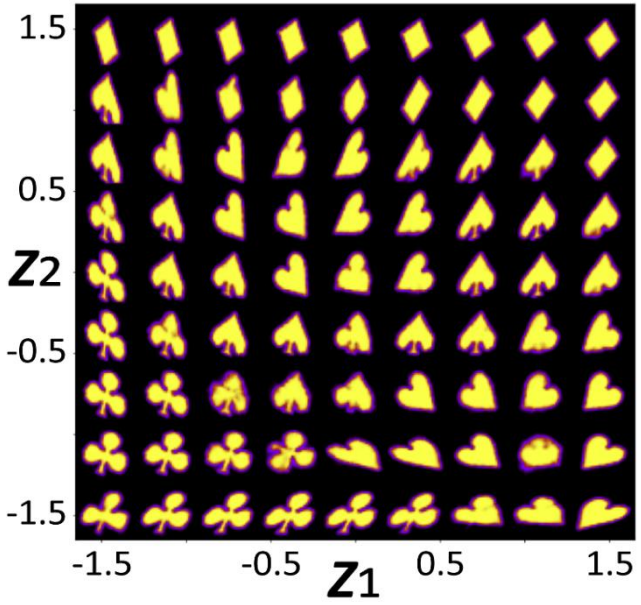



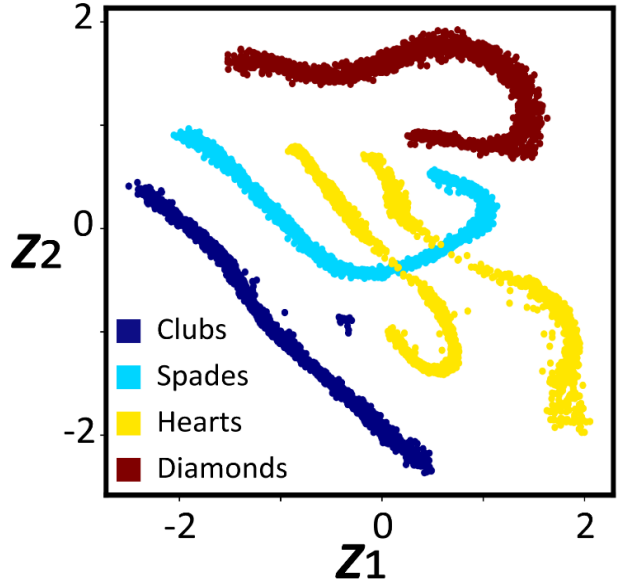



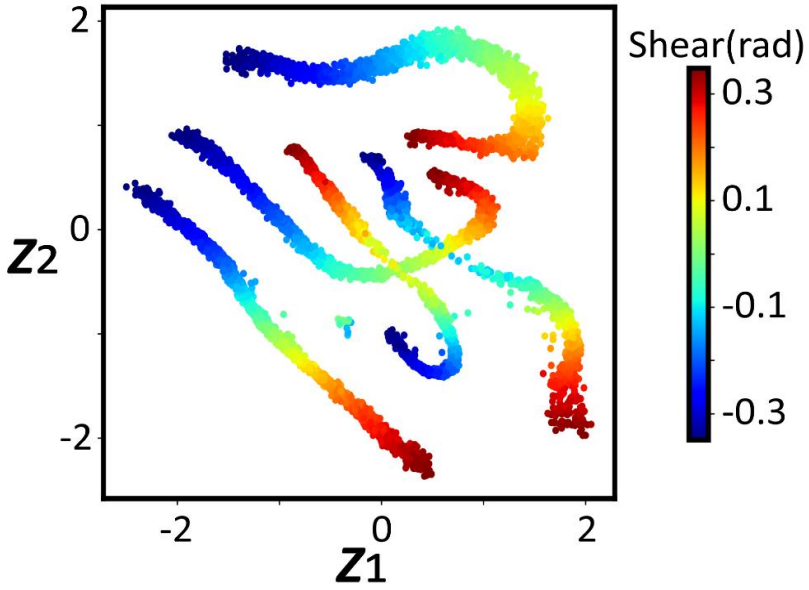



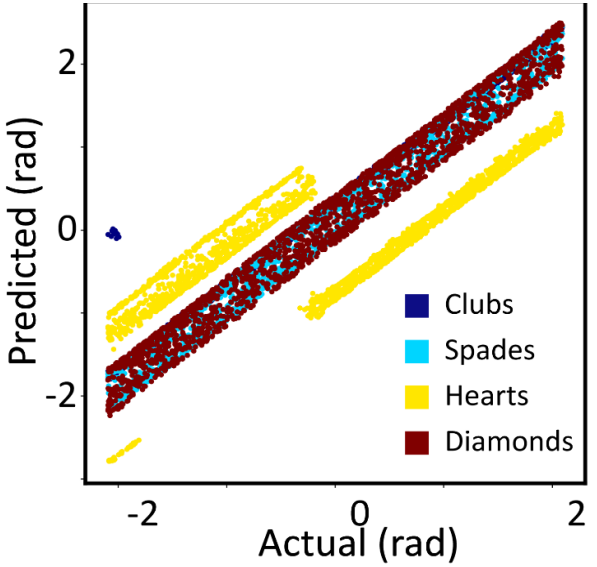

- Clubs
- Spades
- Hearts
- Diamonds

# rVAE on Cards

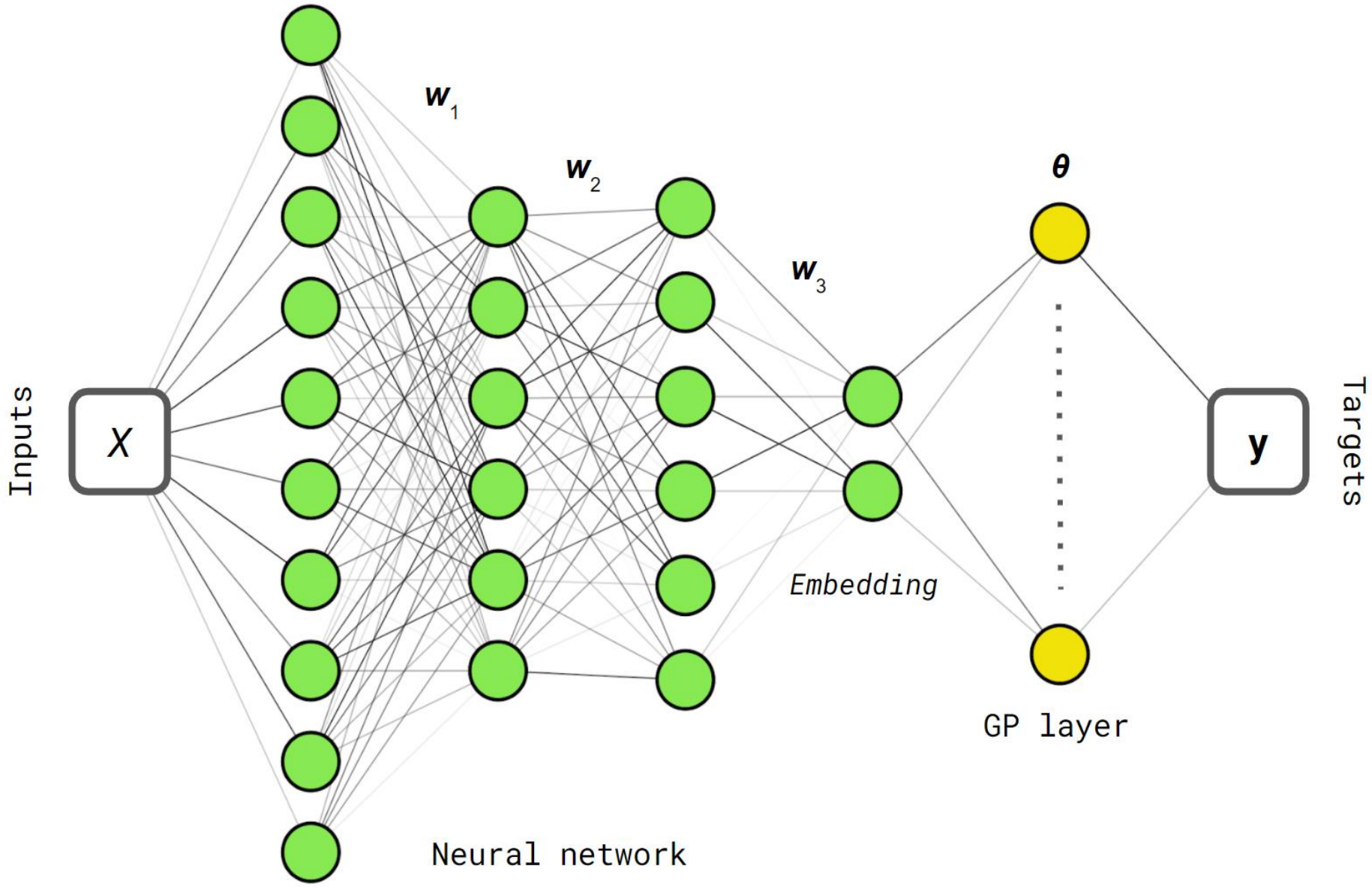


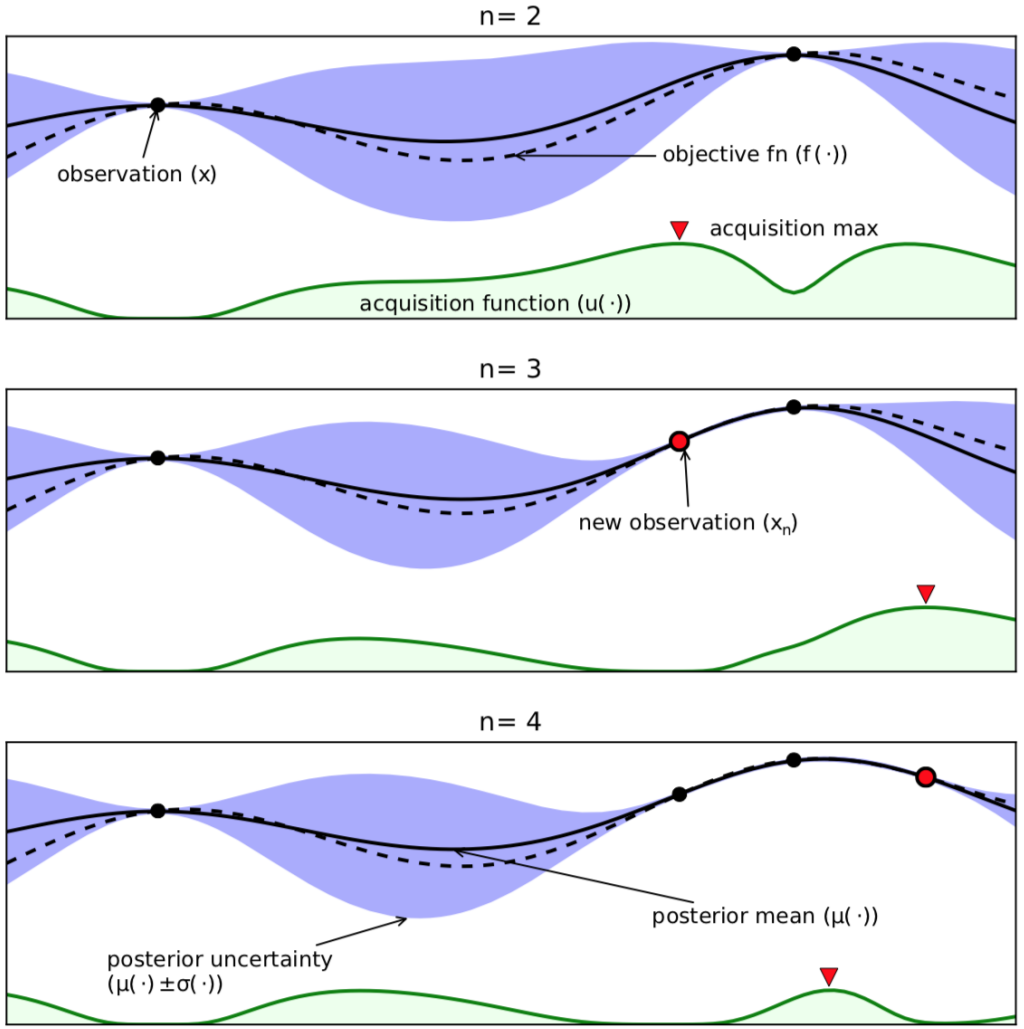



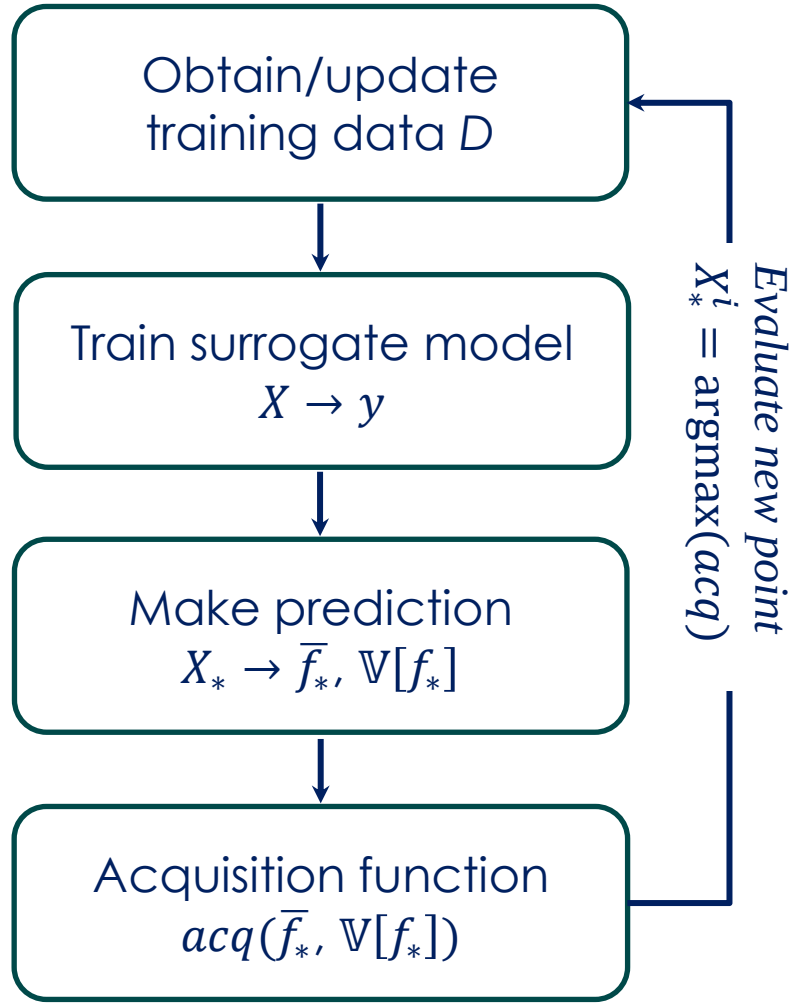
# Deep Kernel Learning



# Bayesian Optimization




$X, y$: (sparse) Training data
$X_*$: New (not yet evaluated) points

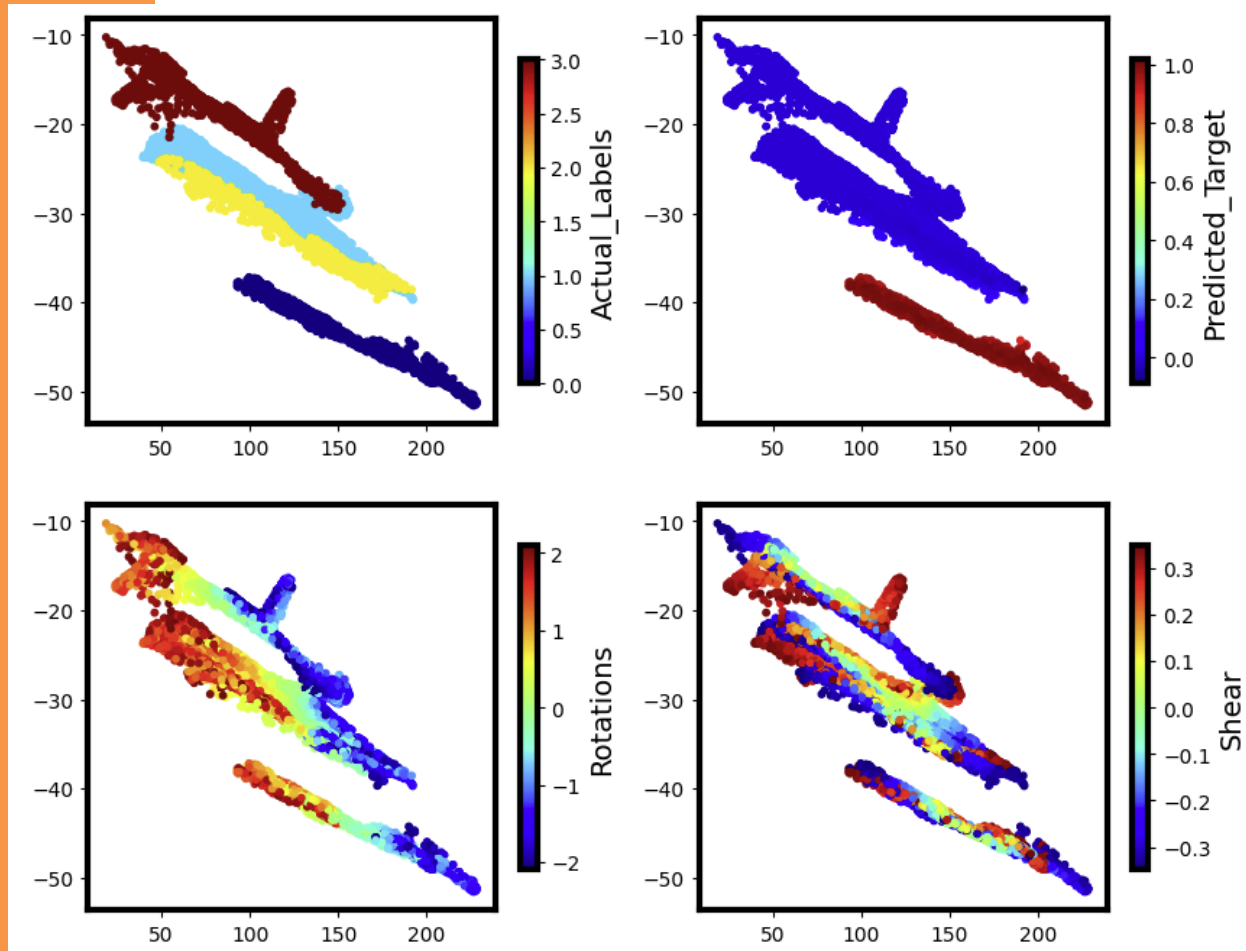


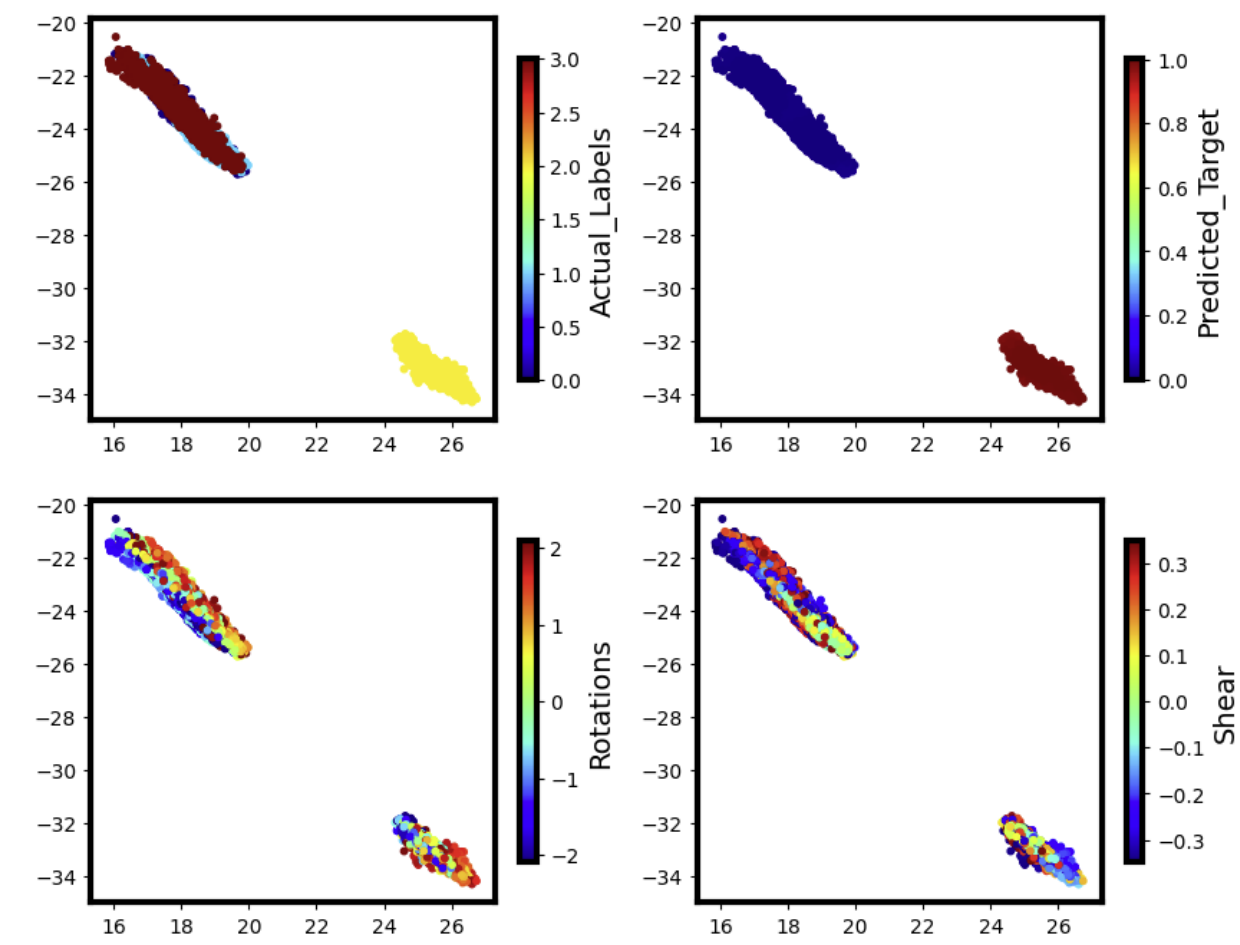

N. de Freitas et al., Taking the Human Out of the Loop: A Review of Bayesian Optimization , *Proceedings of the IEEE* **104**, 148 (2015)

# DKL to predict labels

## Clubs

## Hearts

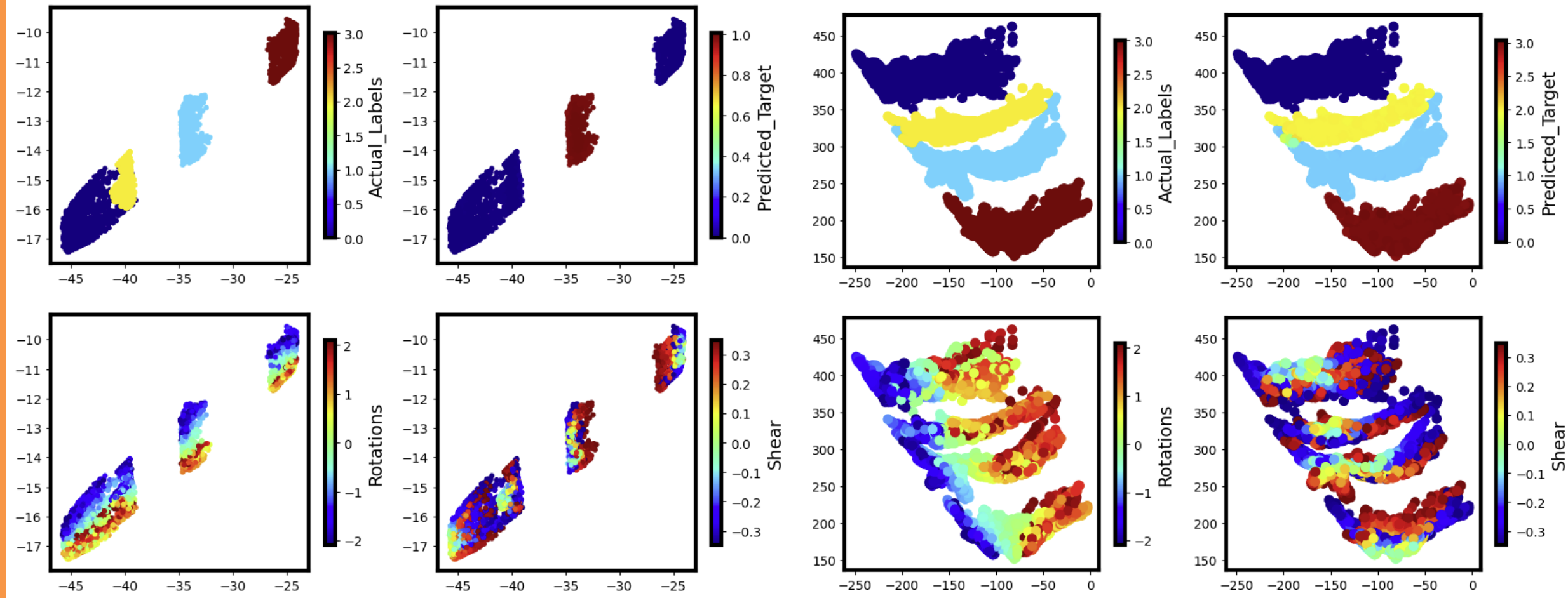


# DKL to predict labels

## Spades

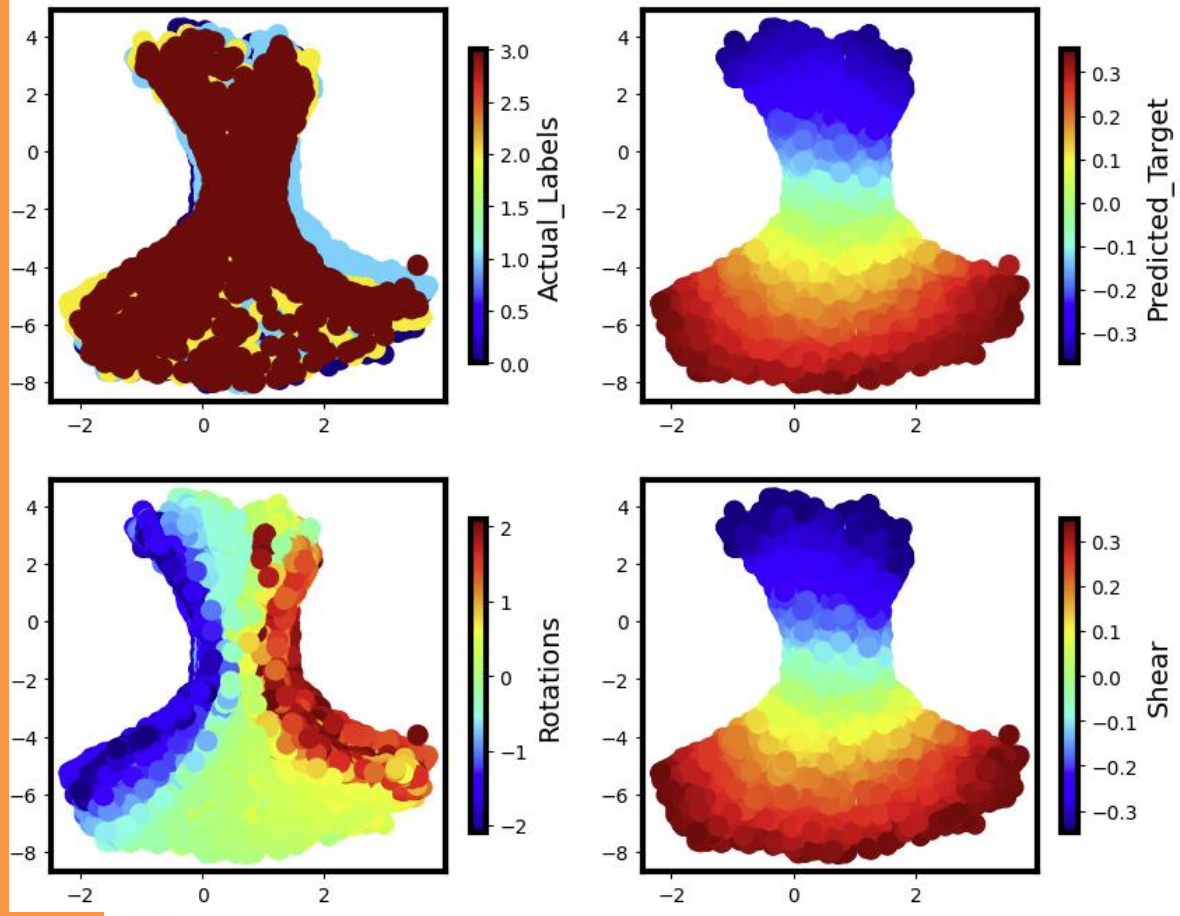
## All suits



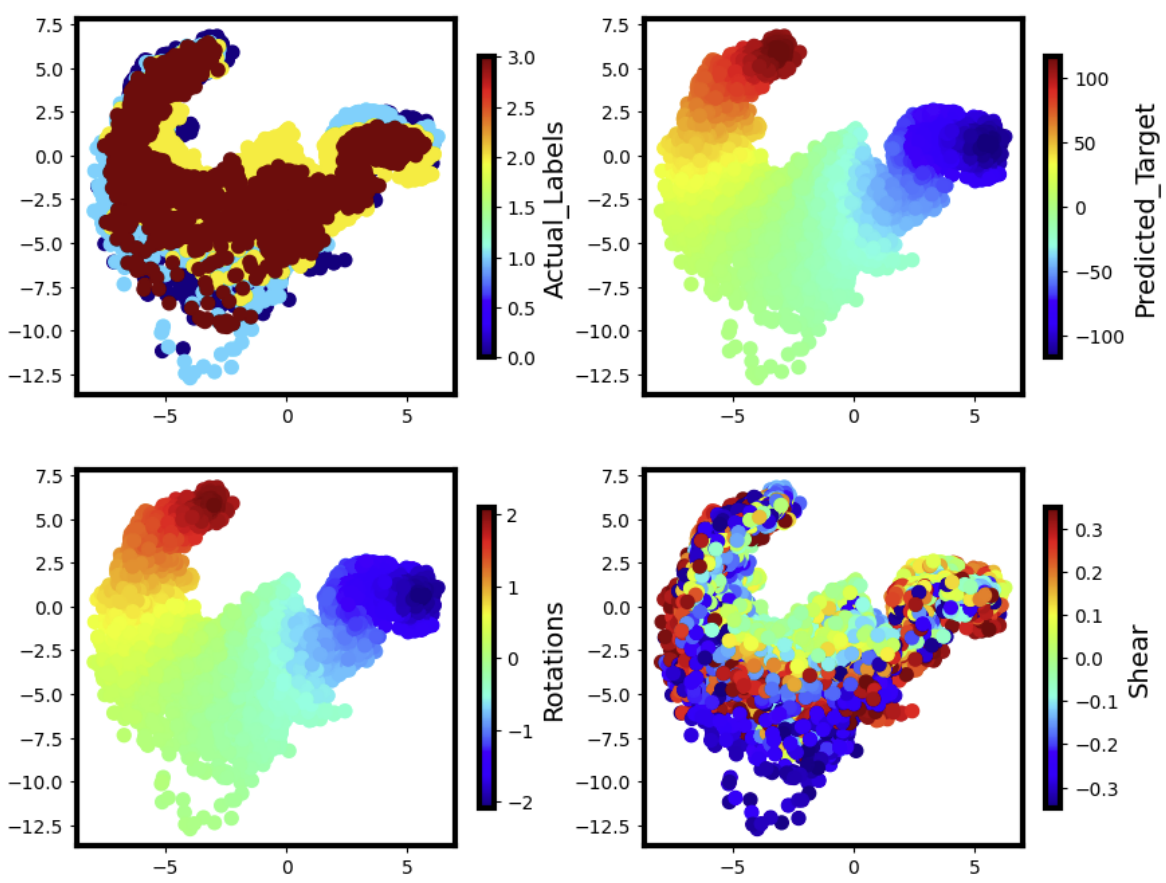

The DKL clearly forms the manifold based on the label!

# DKL to predict continuous target function

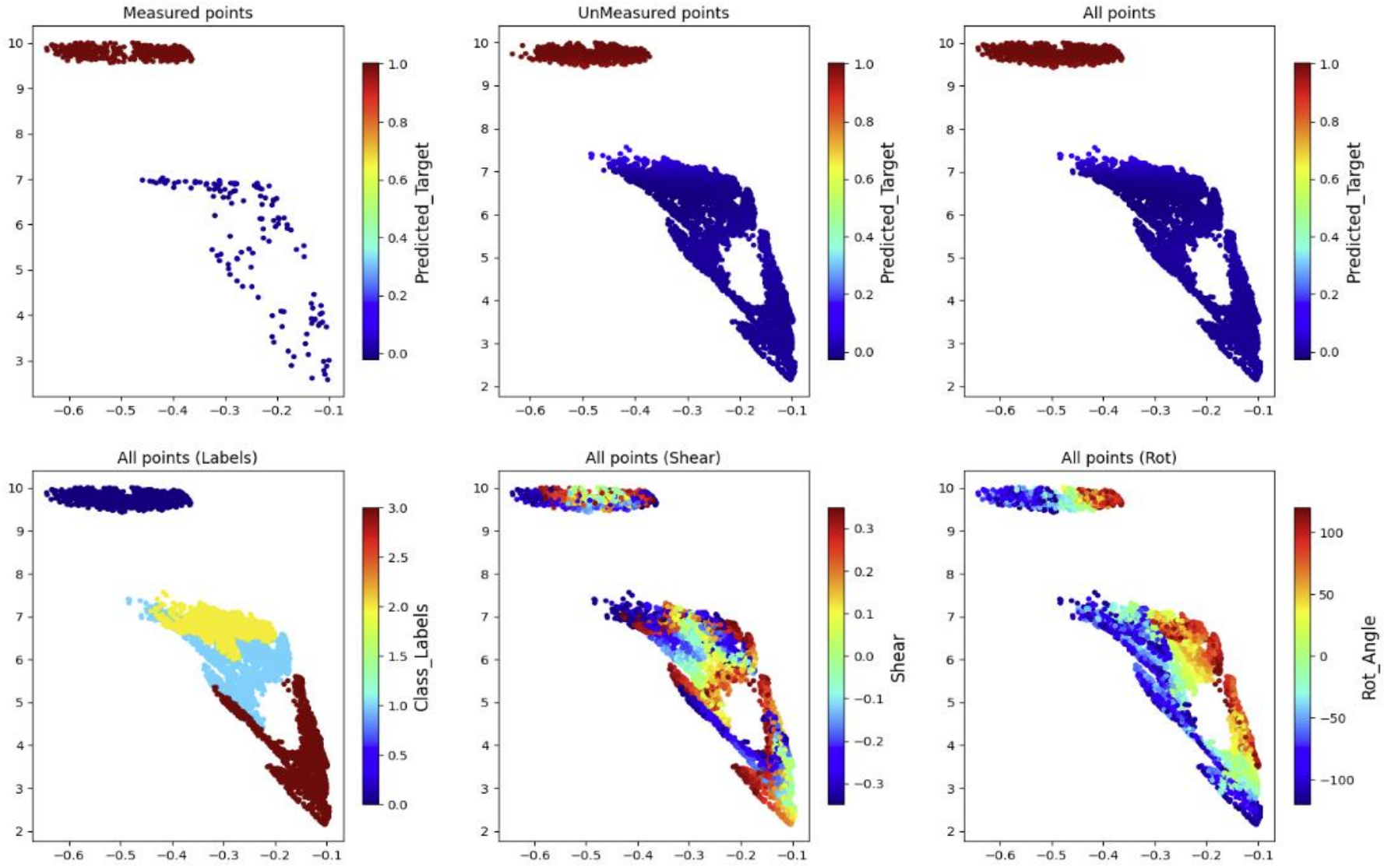
## Shear




## Rotations

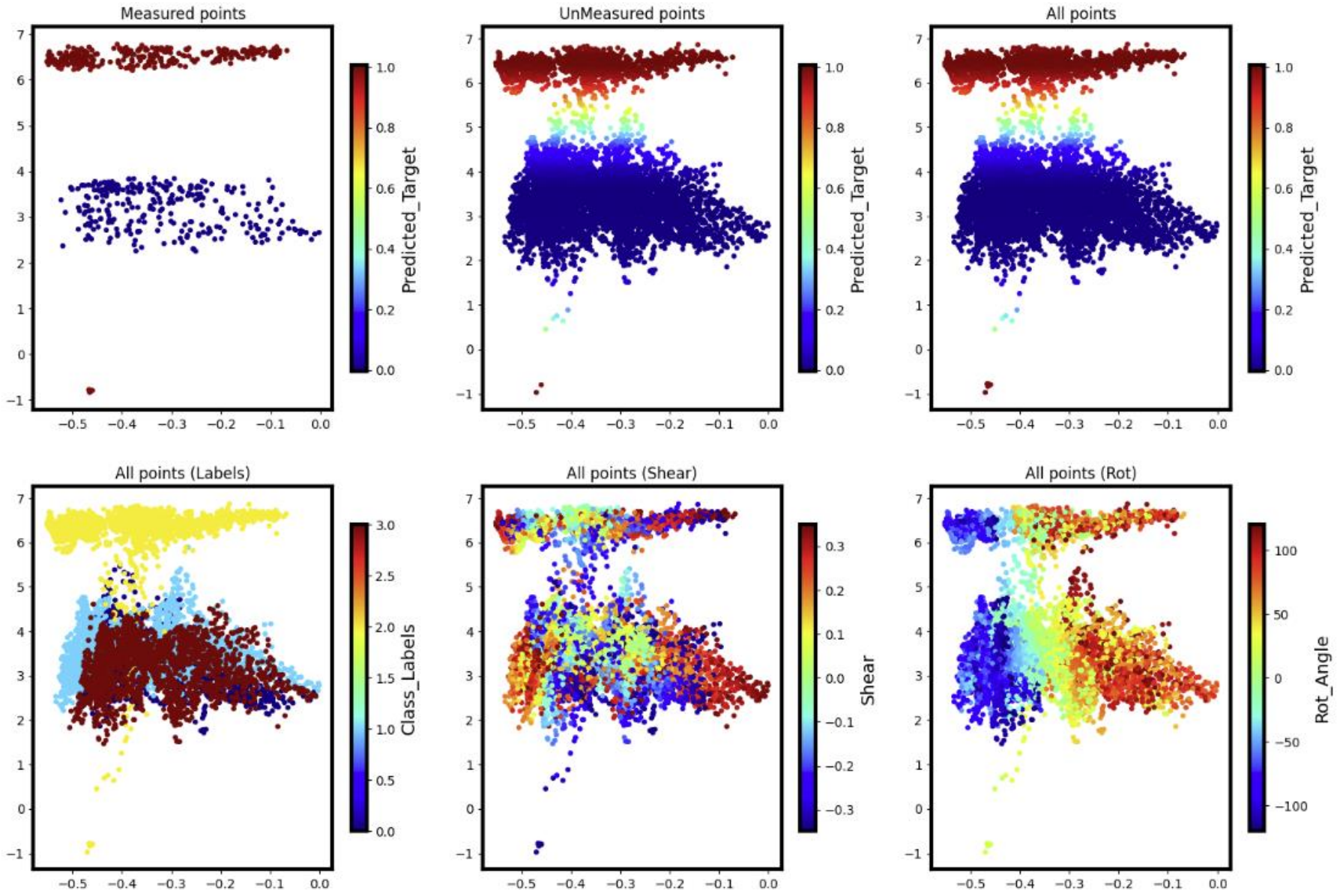


# DKL BO: Active Learning

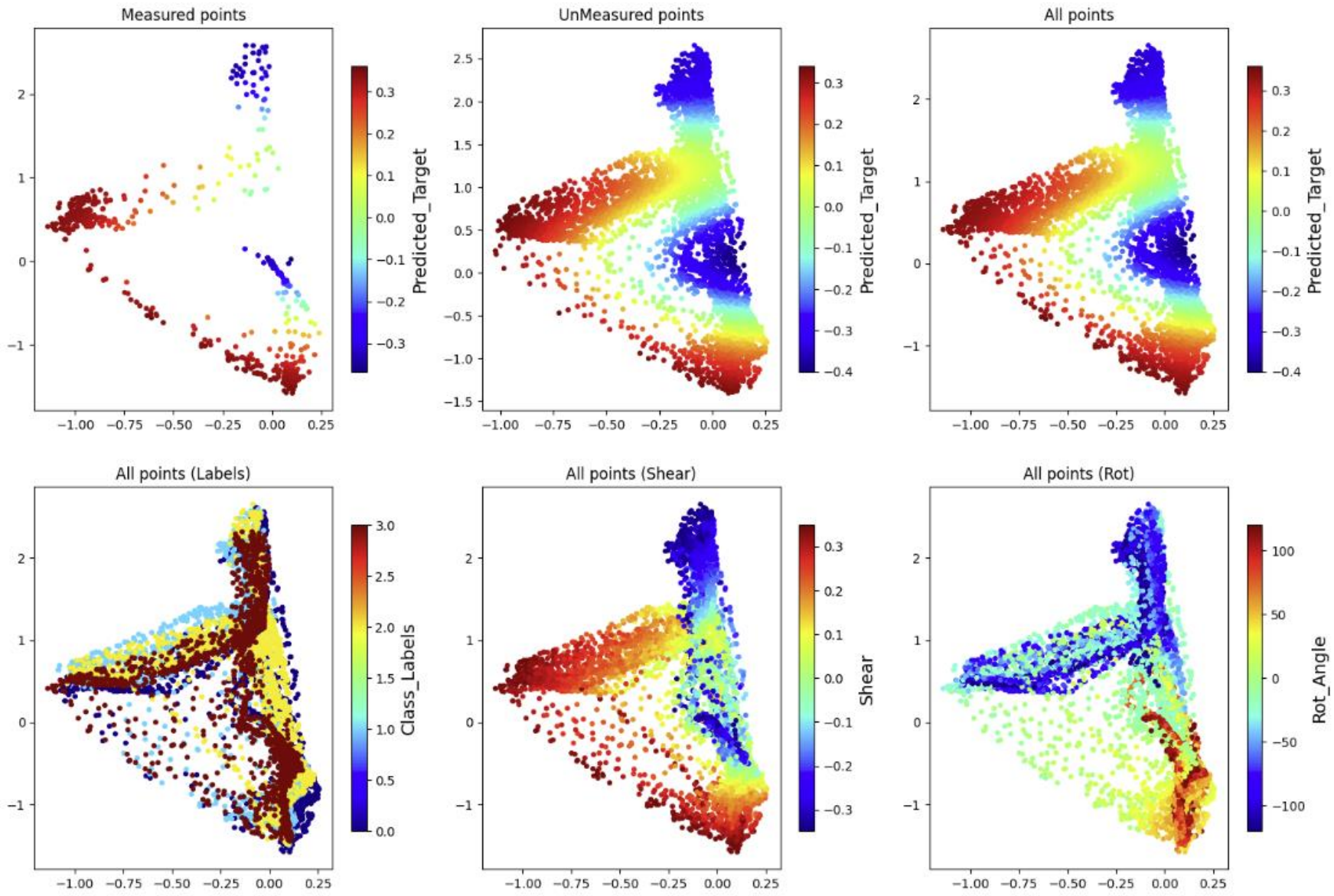



- 100 initialization points and then BO explored 500 points subsequently
- Acquisition function: μ + 10σ

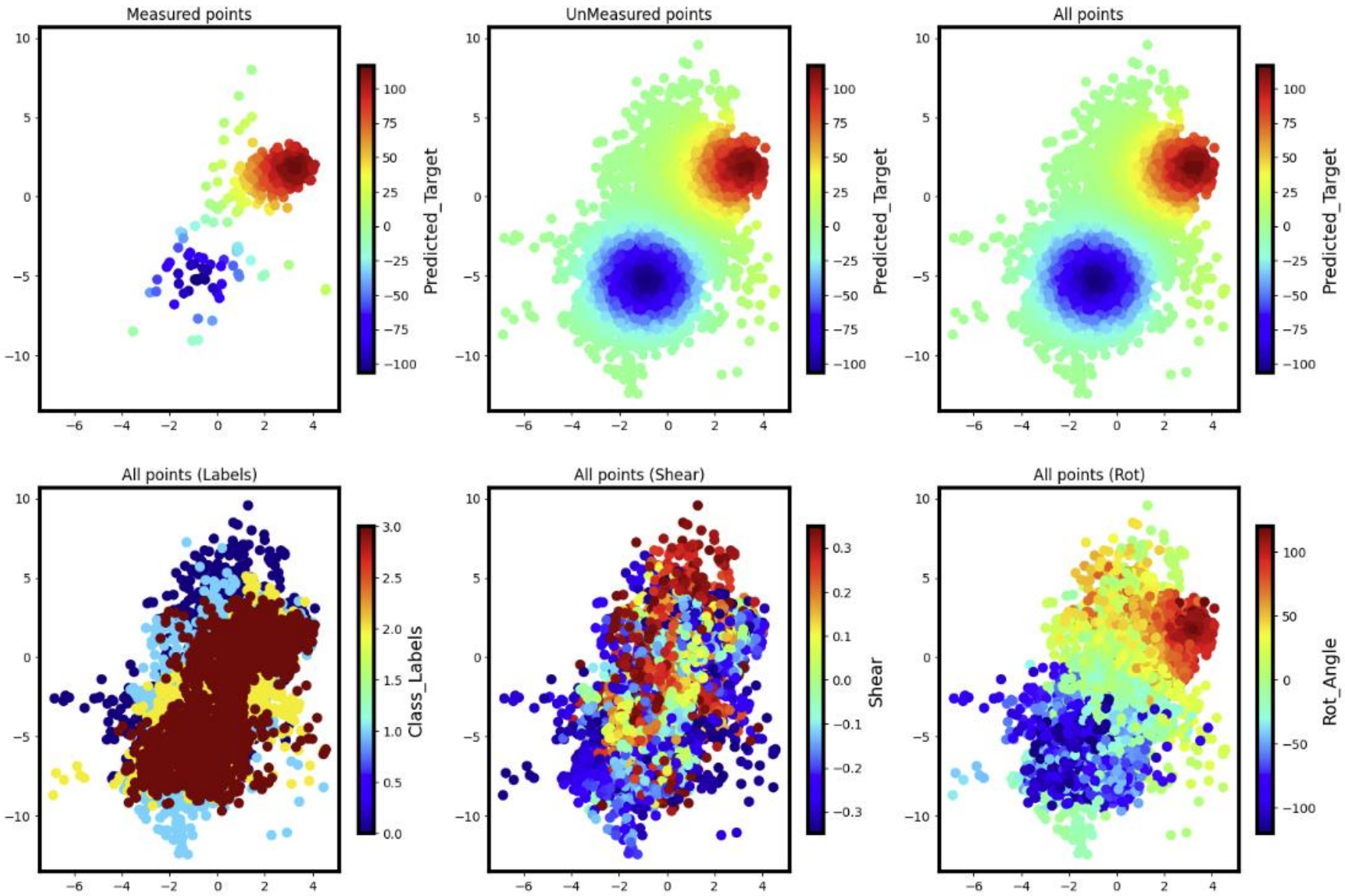
# DKL BO: Hearts



# DKL BO: Shear

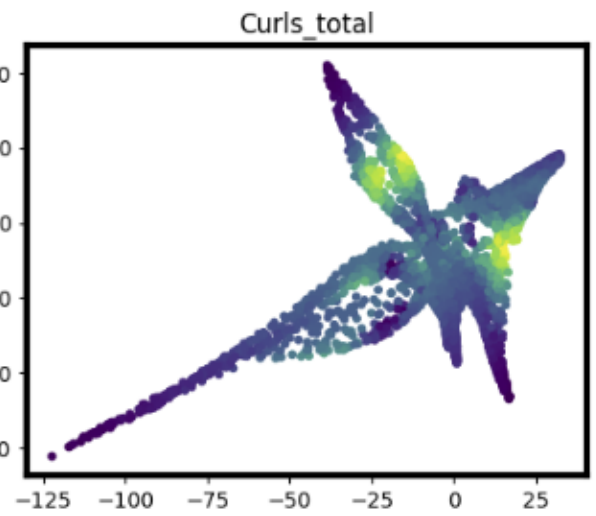

Measured points
UnMeasured points
All points
All points (Labels)
All points (Shear)
All points (Rot)
Predicted_Target
Class_Labels
Shear
Rot_Angle

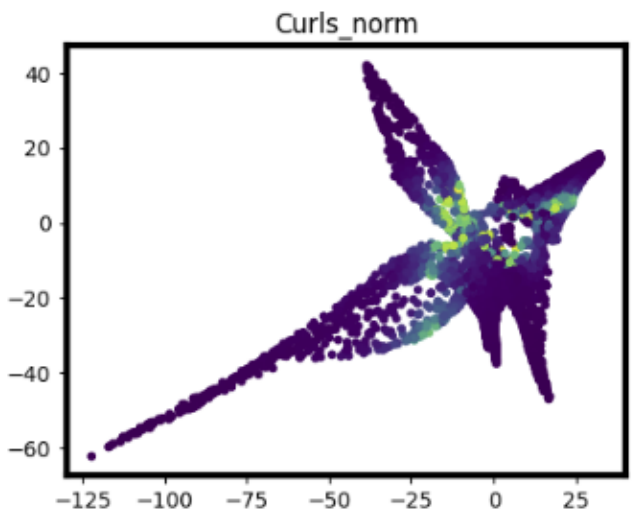
# DKL BO: Rotations

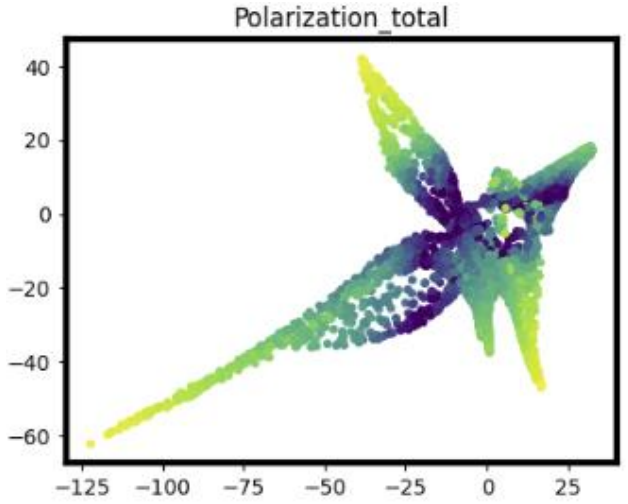


# DKL on FerroSIM: Static

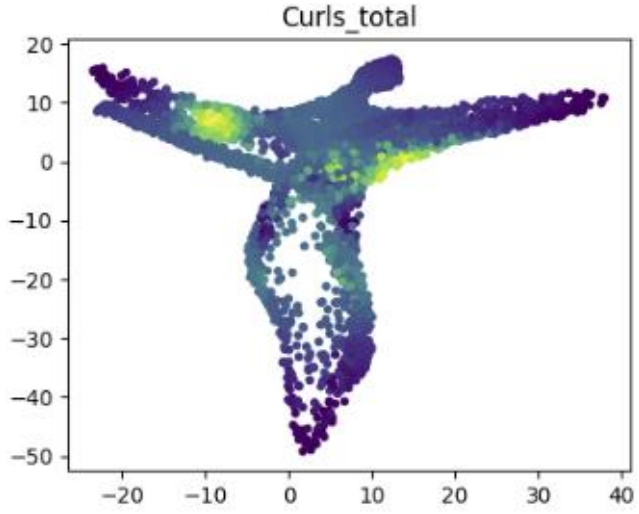
Target function: Curl

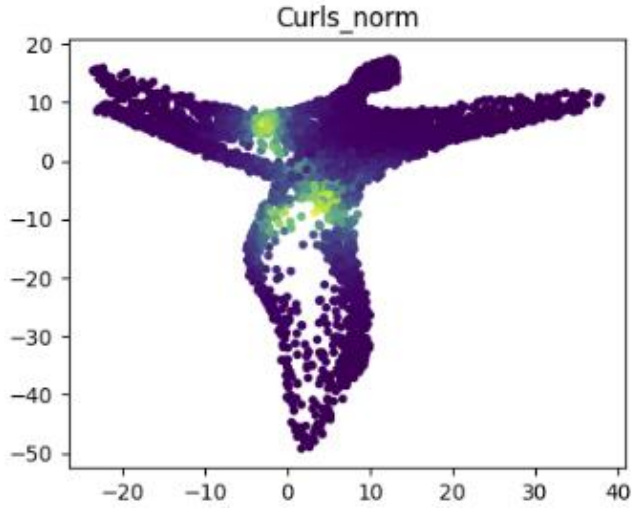


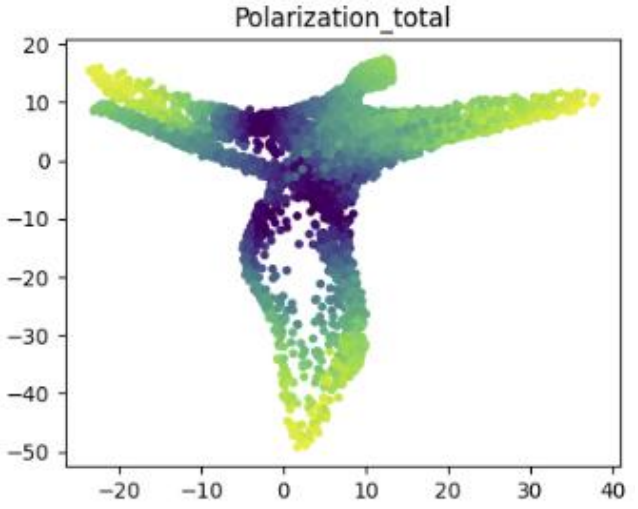




Target function: Normalized curl

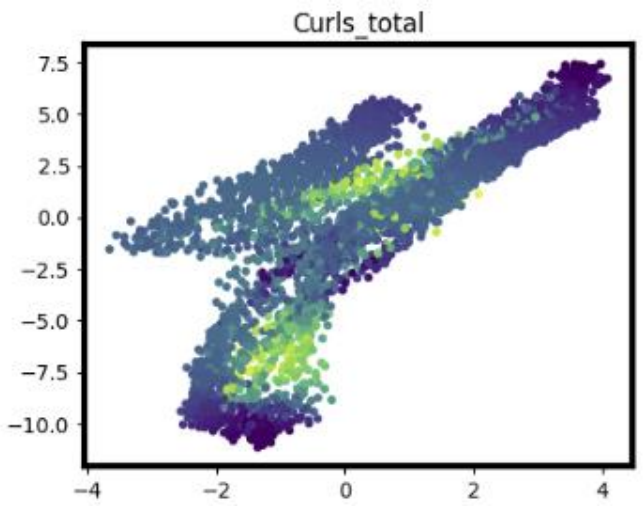


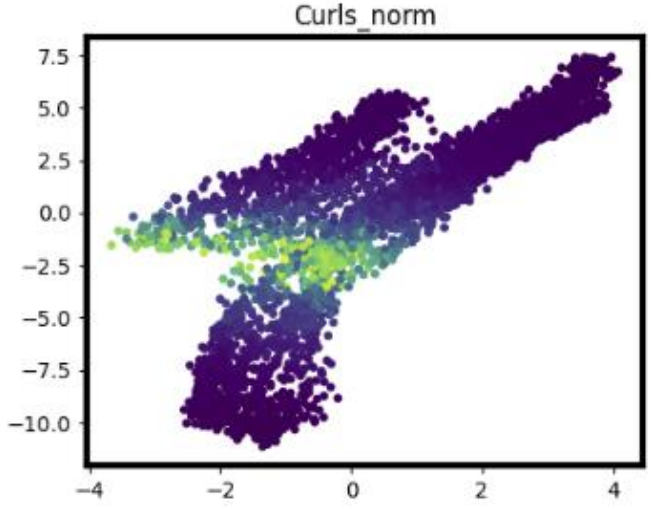



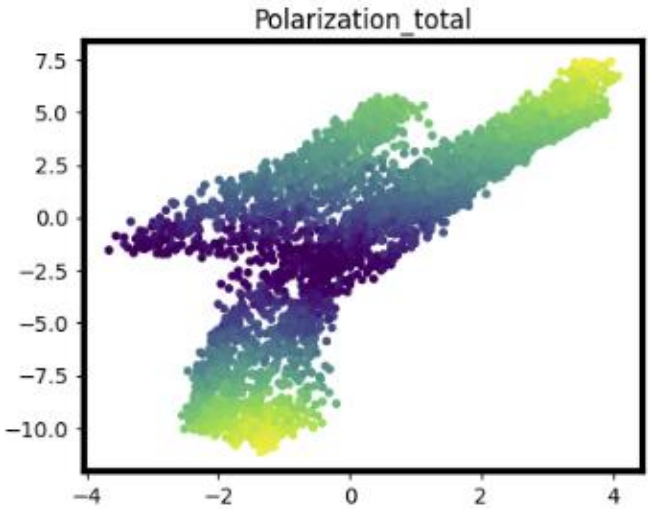



# DKL on FerroSIM: Static

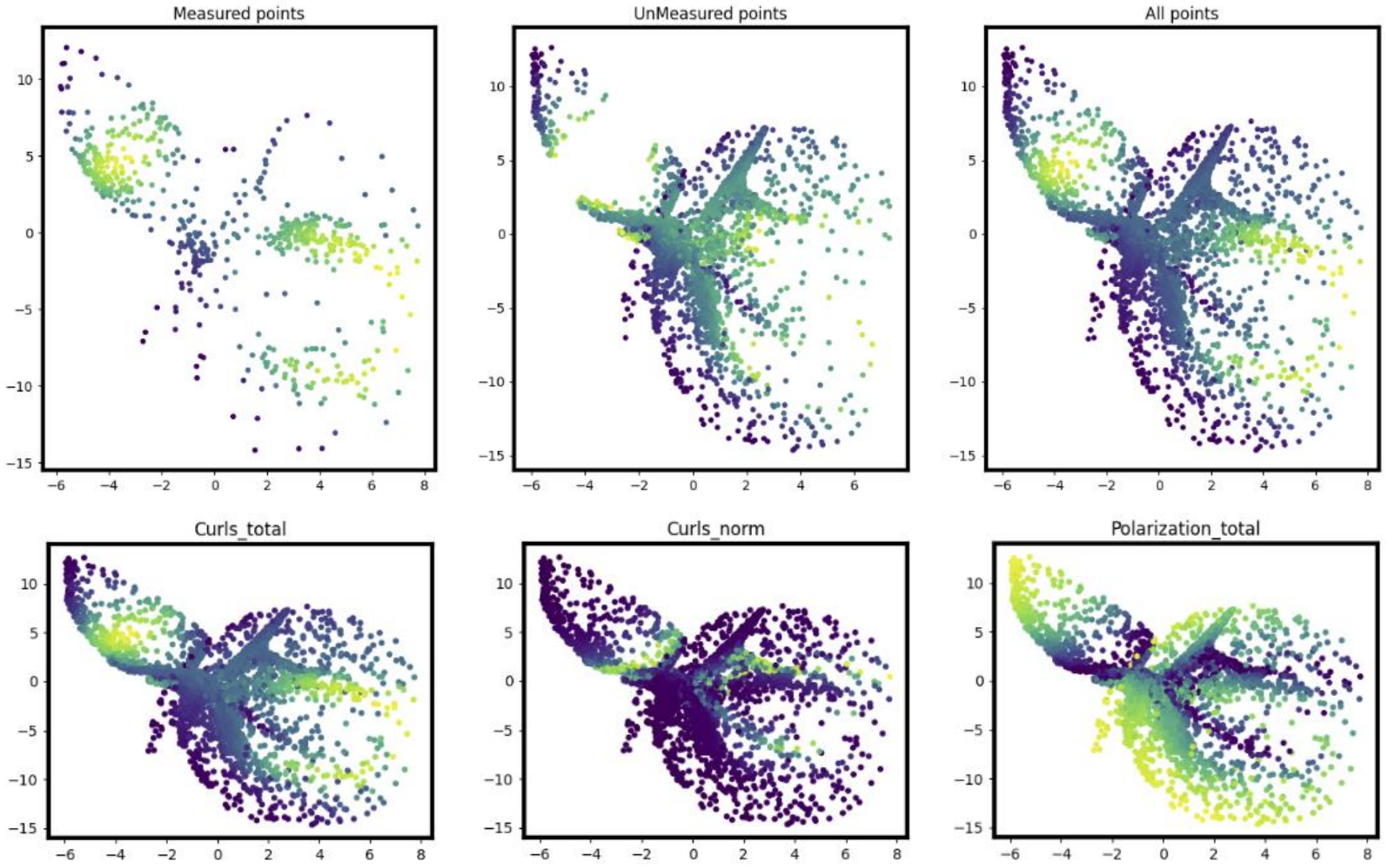
Target function: Polarization



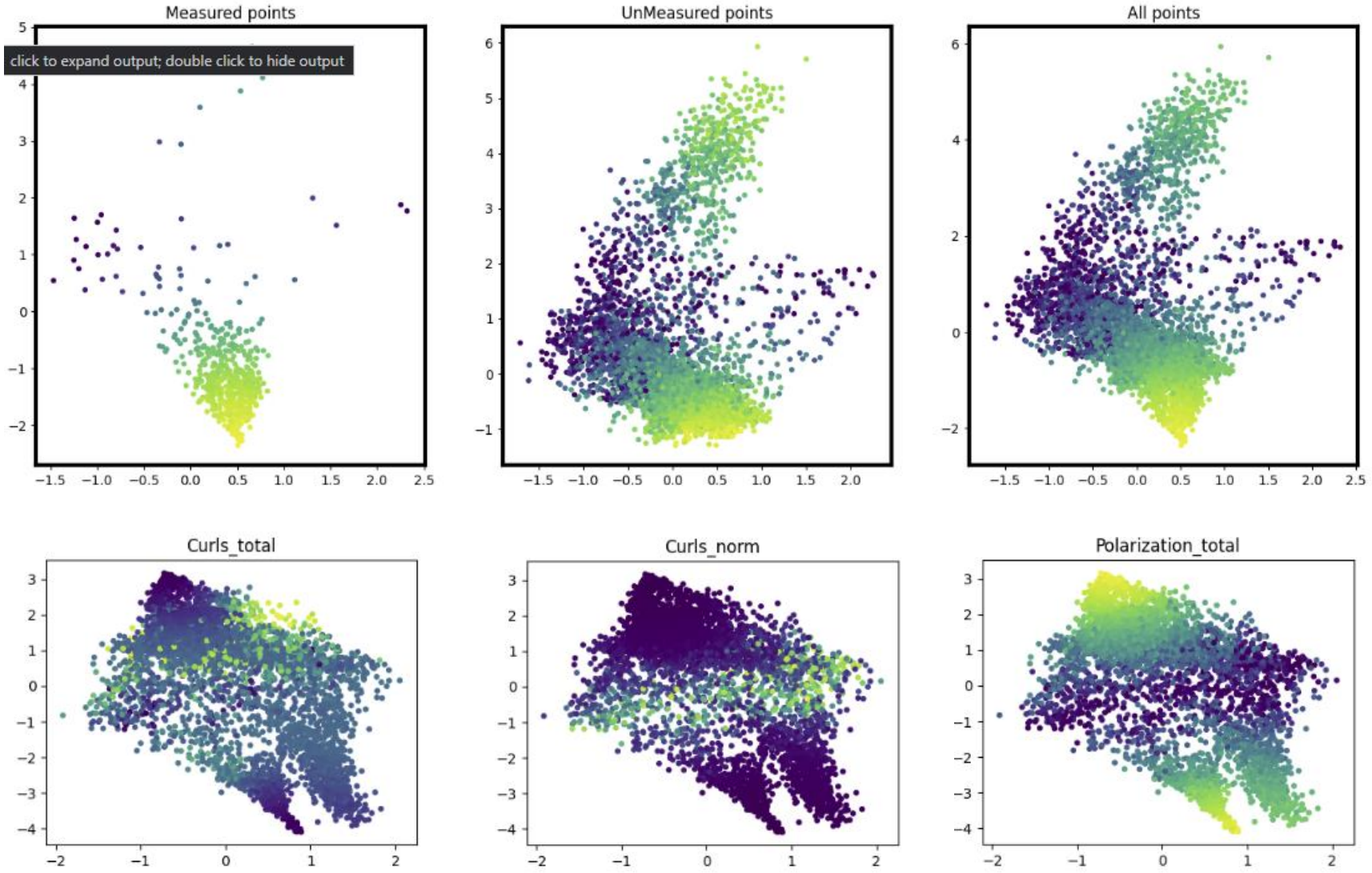



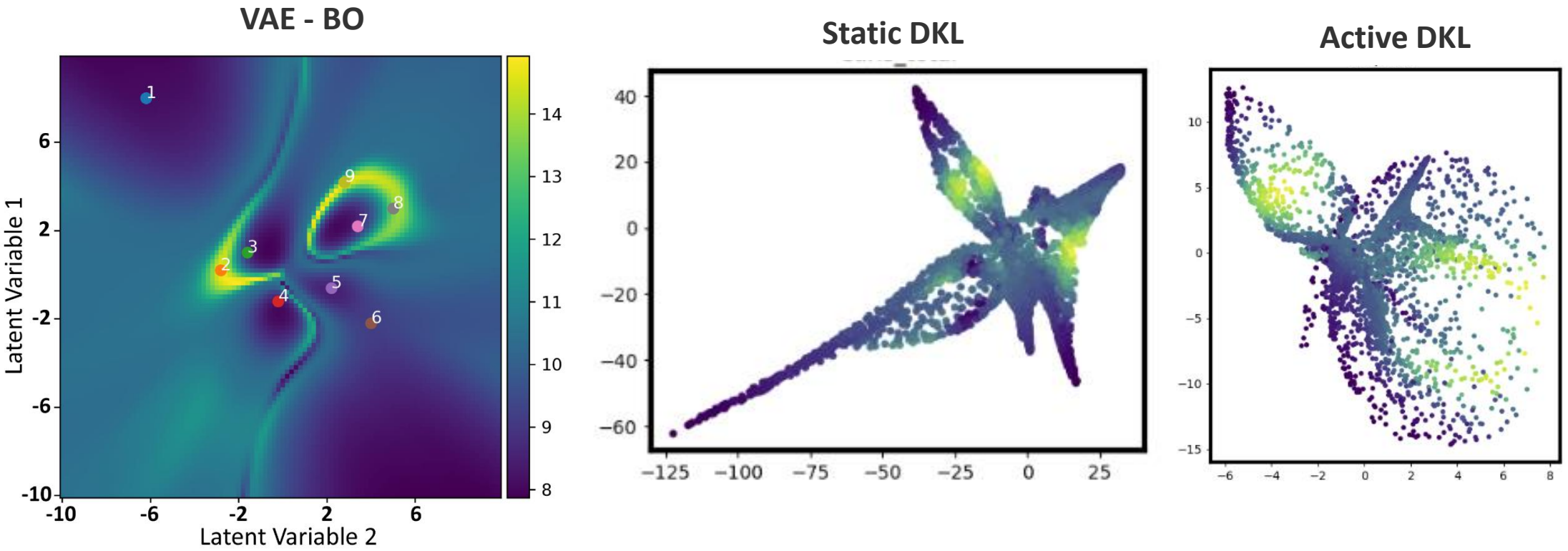
# DKL BO: Active Learning of Curl



# DKL BO: Active Learning of Polarization



# Comparing VAE BO, Static DKL, and Active DKL




**Summary:**

- Manifold structure determines how fast can the unsupervised or active learning work
- For VAEs, the latent structure is determined by the data only. Sometimes property are forming convenient manifolds, most of the time not.
- Static DKL forms much better organized manifolds
- … Active learning produces best manifolds!

# Colab: DKL Molecules

# Two modes of operations

**Structural imaging (Cheap)**

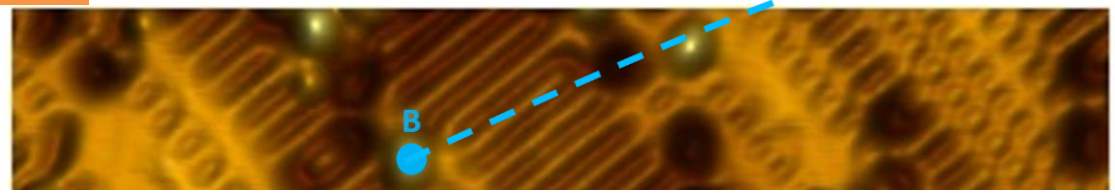
Topography in STM, amplitude/phase in SPM, (HA)ADF-image in STEM, etc.

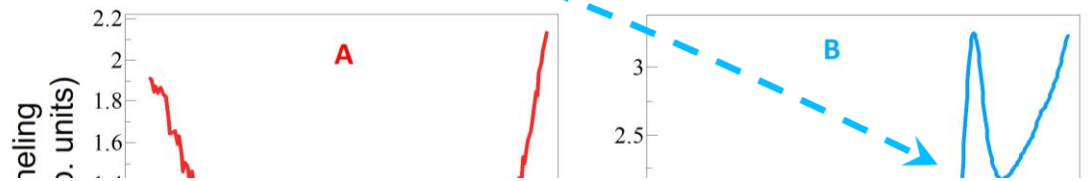
These are FAST measurements
(from seconds to minutes)

**Functional imaging (Costly)**

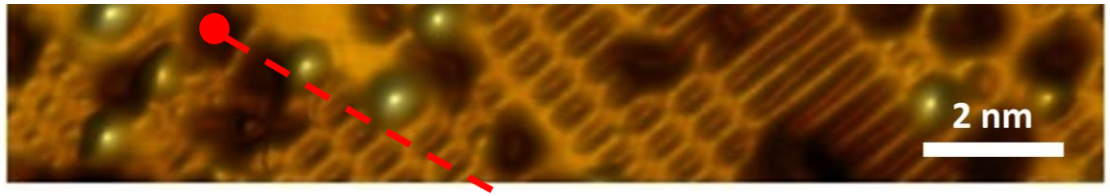
Scanning tunneling spectroscopy (STS), polarization loops in SPM, EELS in STEM, etc.

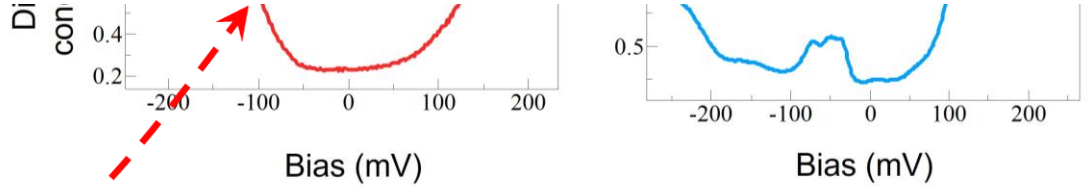
These are SLOW and/or DESTRUCTIVE measurements
(from minutes to days)



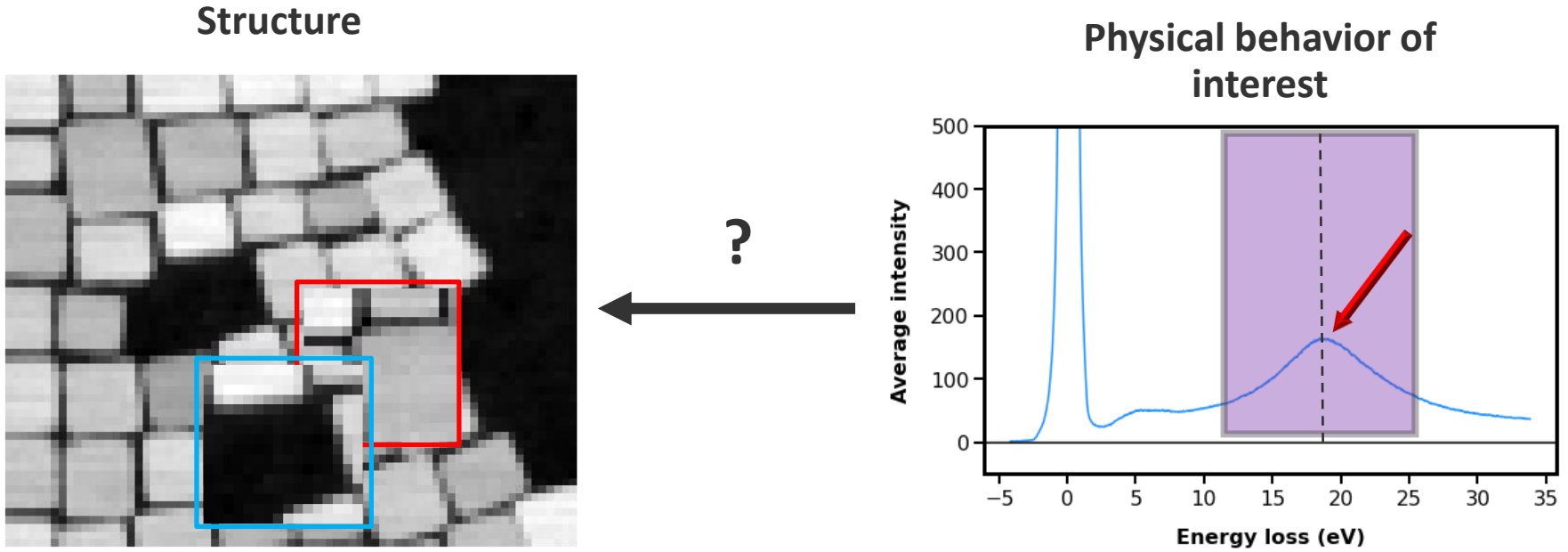

**Can we use structural information to guide functional measurements and in the process learn structure-property relationships?**



# Physics discovery in active experiments

- Suppose we know what physical behavior/property we are interested in (superconductivity, ferroelectric switching, plasmonic modes, etc.)
- This behavior is encoded in spectra that we can measure everywhere in the sample (size of superconducting gap, polarization loop area, peak intensity, etc.)
- We want to identify (local) structural features where this behavior is maximized/minimized
- We want to achieve this with as few measurements as possible (**< 5% of the entire grid**)

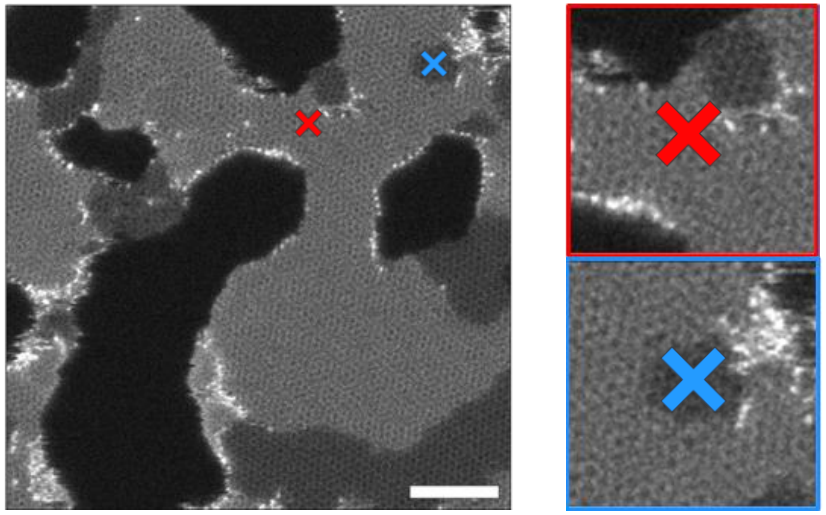


# Deep Kernel Learning: Step 1

A1

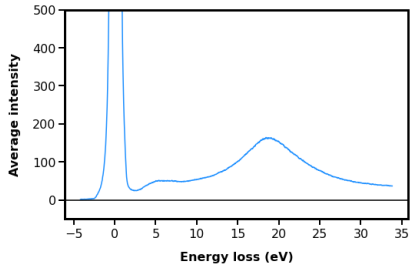
Full image

Image patches



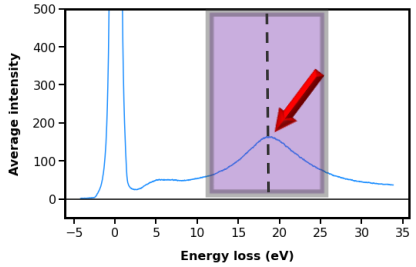
A2

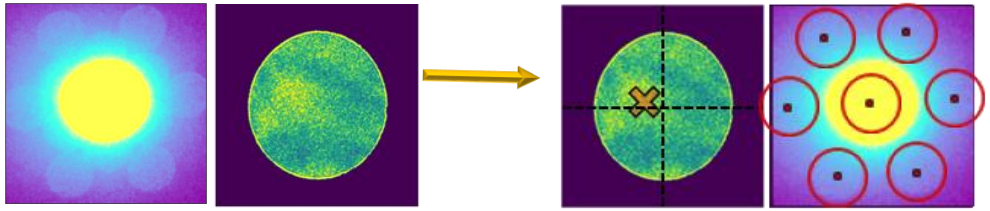
**Measurement (1D or 2D)**




**Scalar target**

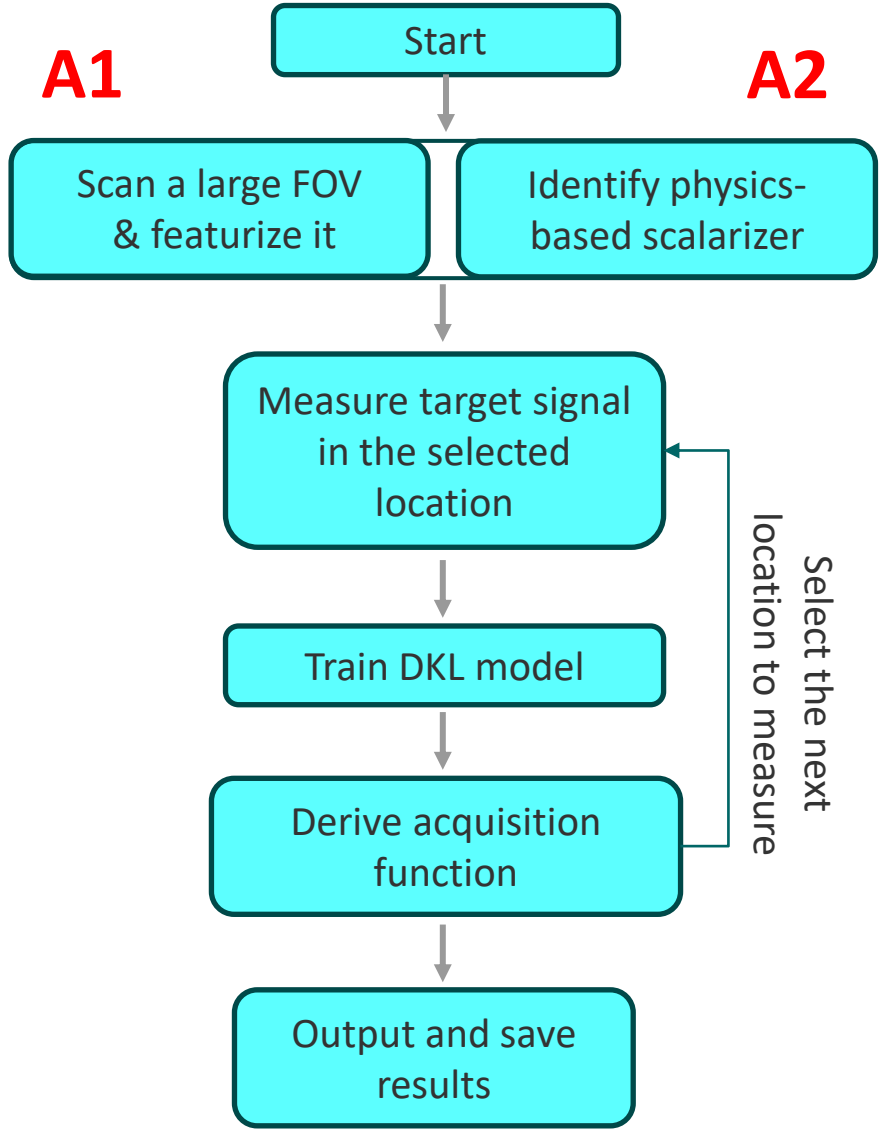
Peak maximum



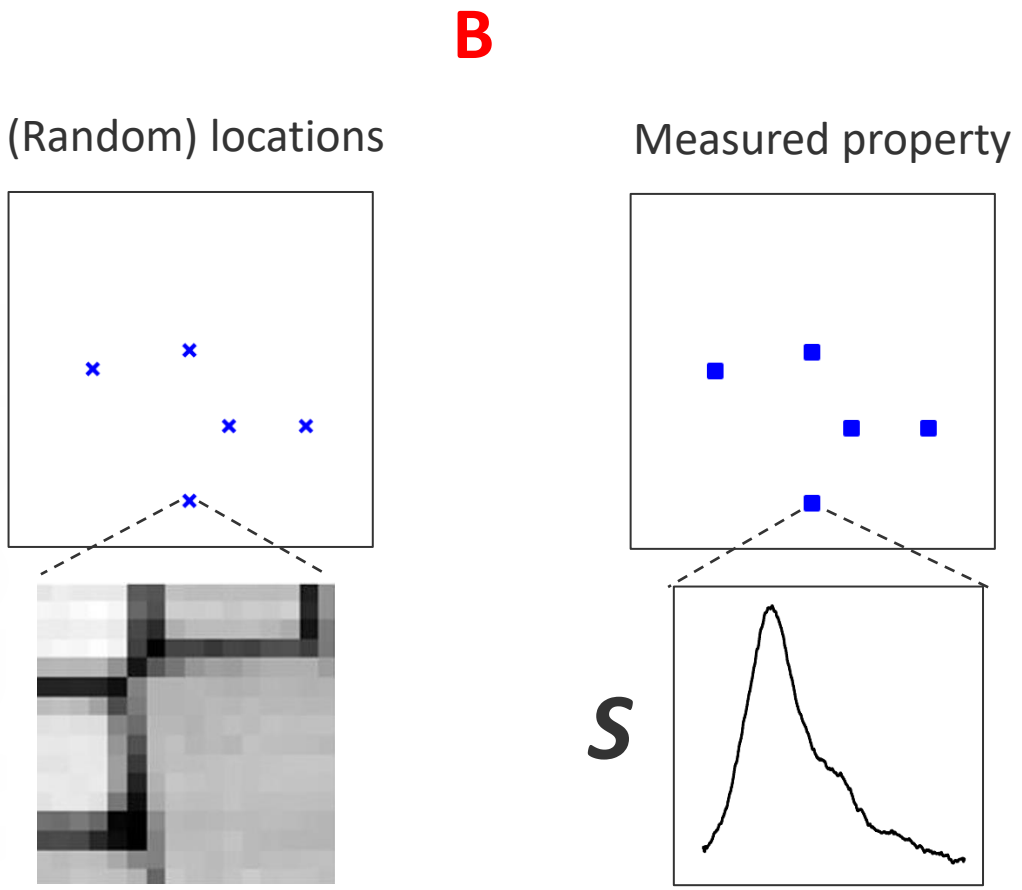



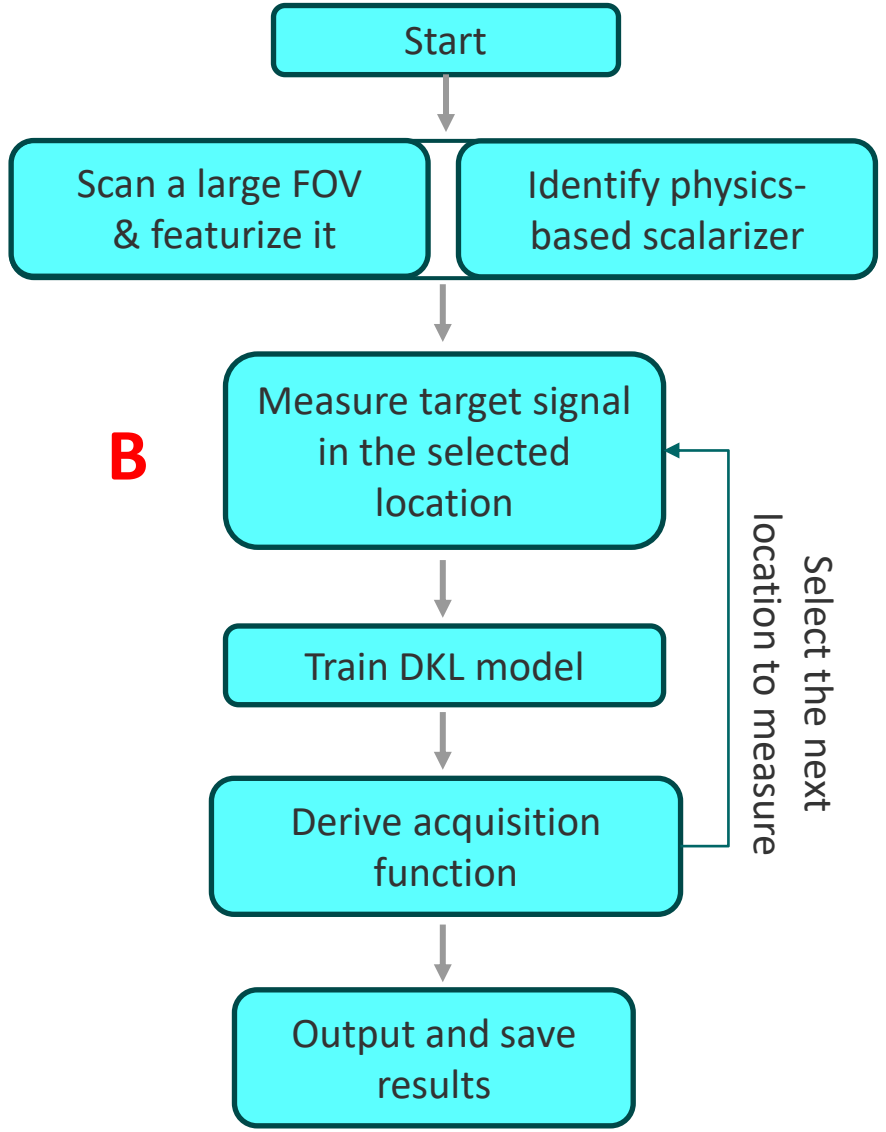

A1

A2

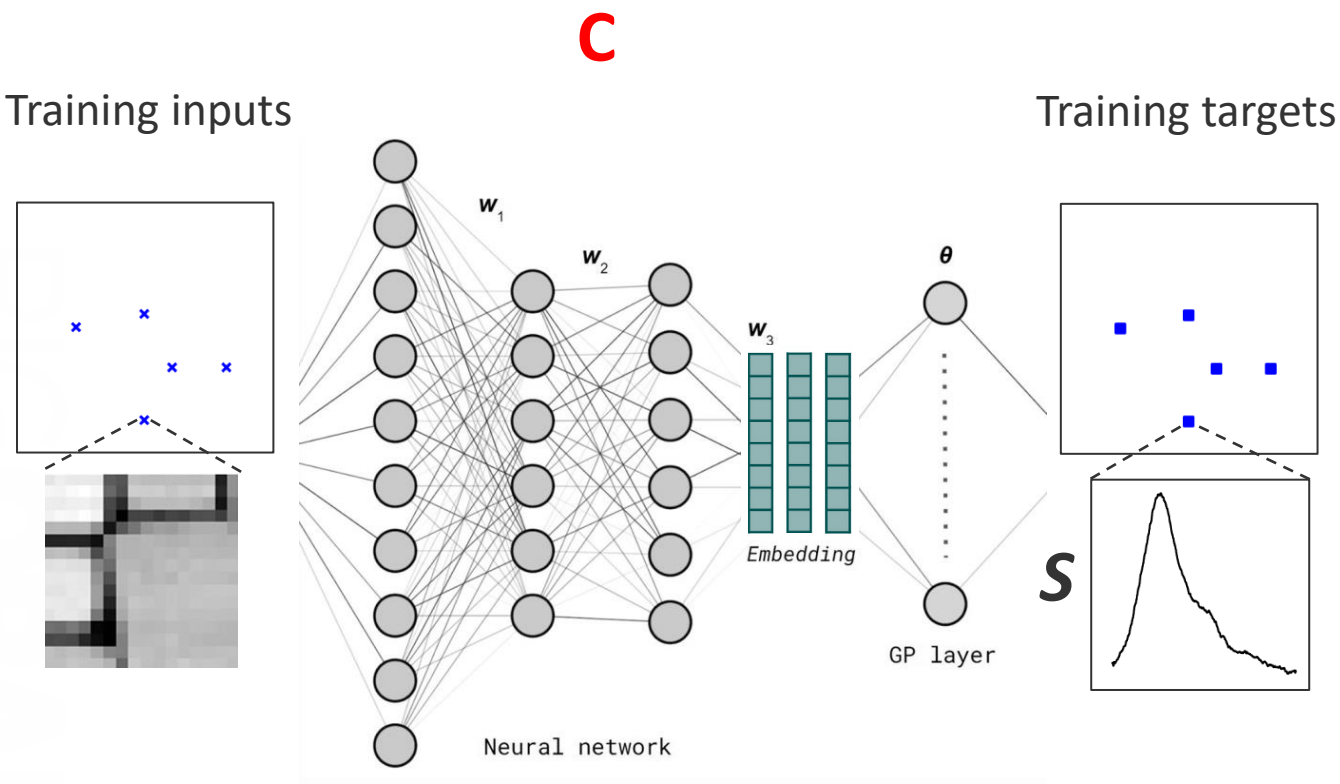


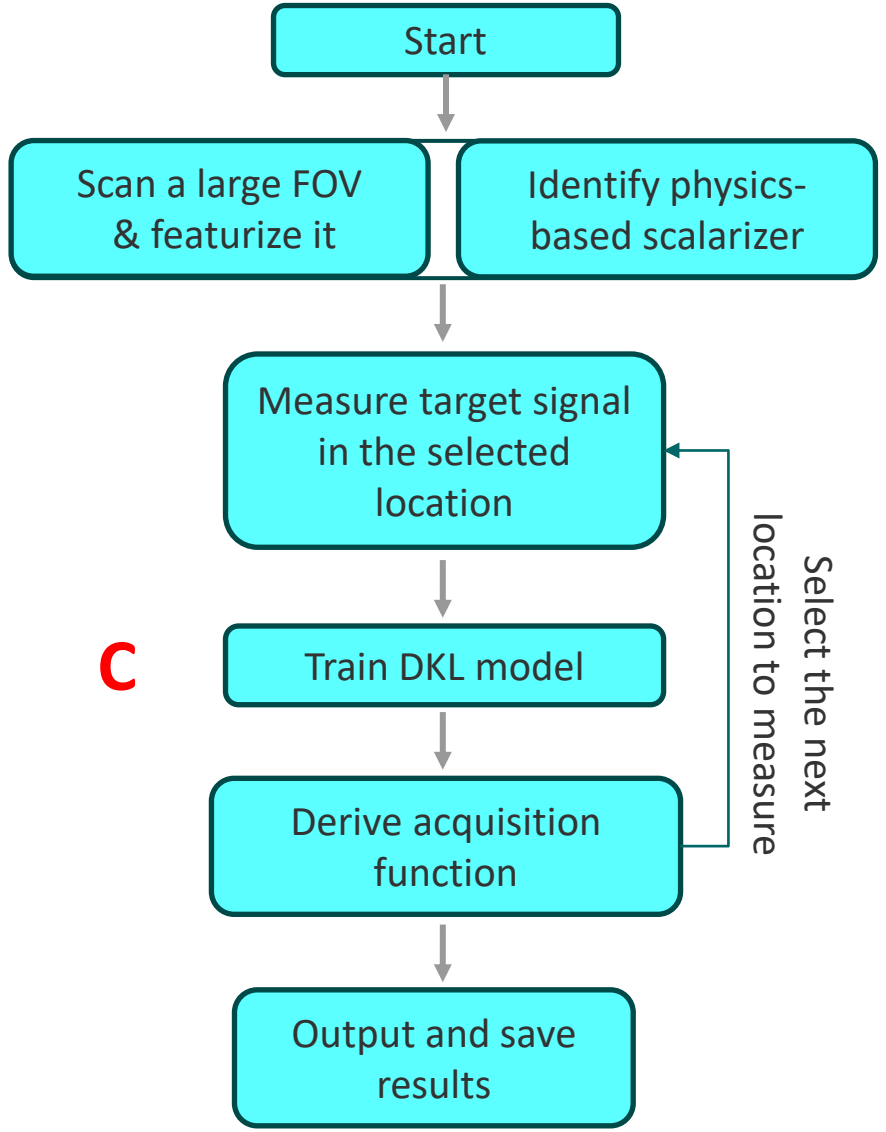
# Deep Kernel Learning: Step 2



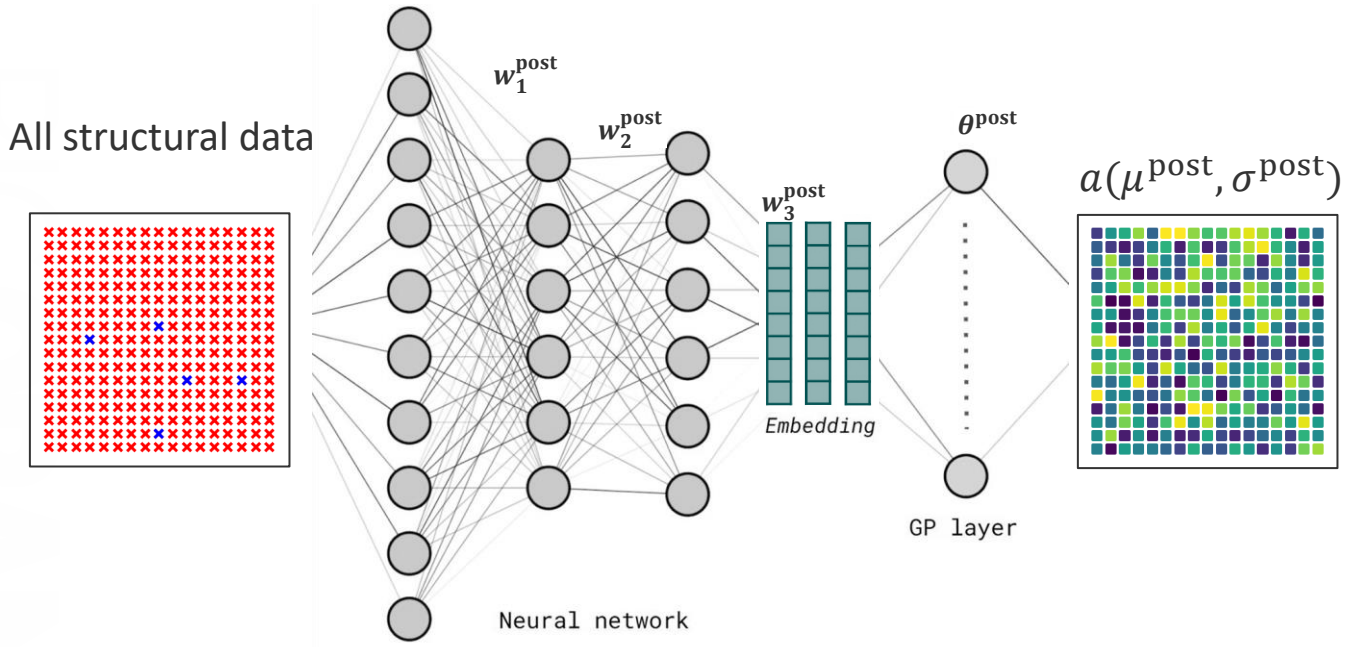



# Deep Kernel Learning: Step 3



# Deep Kernel Learning: Going Active

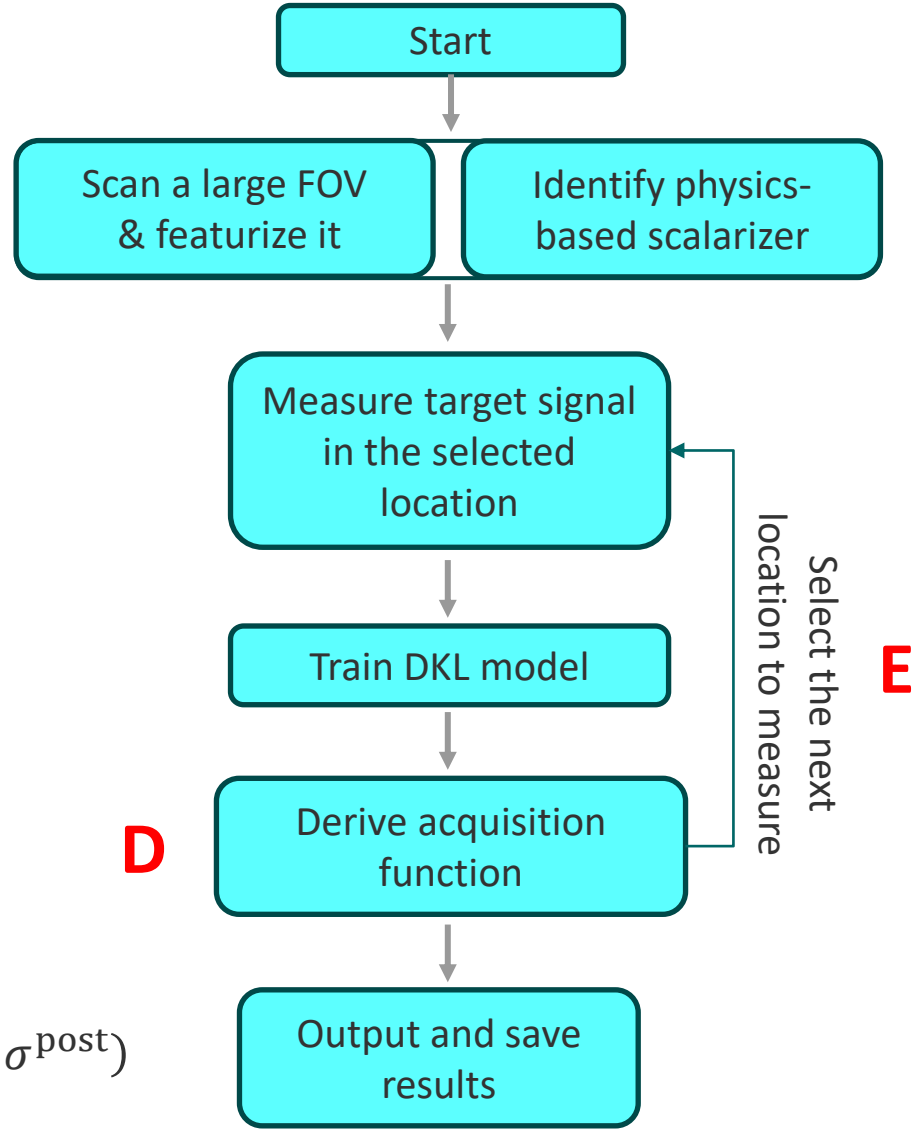



D

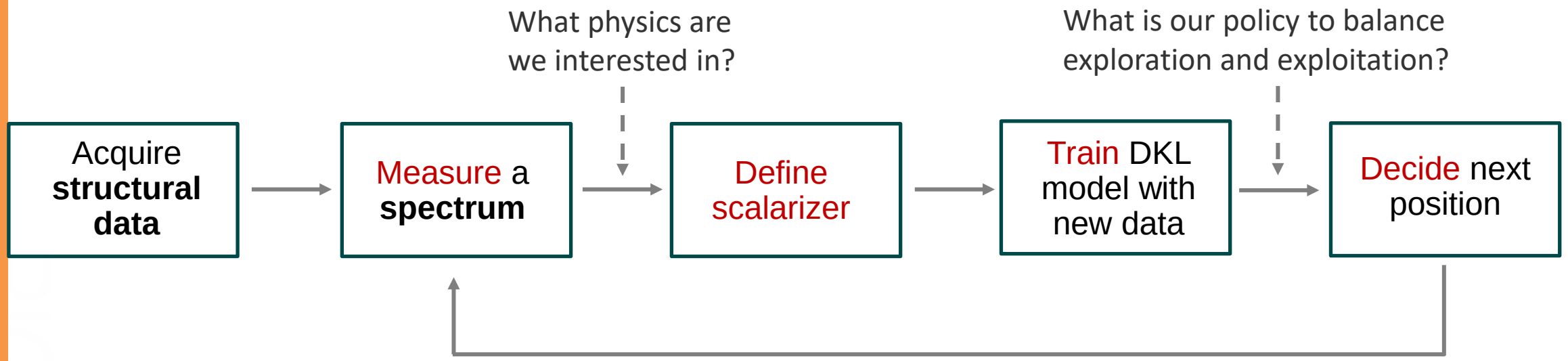
E

Next point to measure:

$$x_{next} = \arg\max_{\mathbf{x}} a(\mu^{post}, \sigma^{post})$$



# Deep Kernel Learning based BO

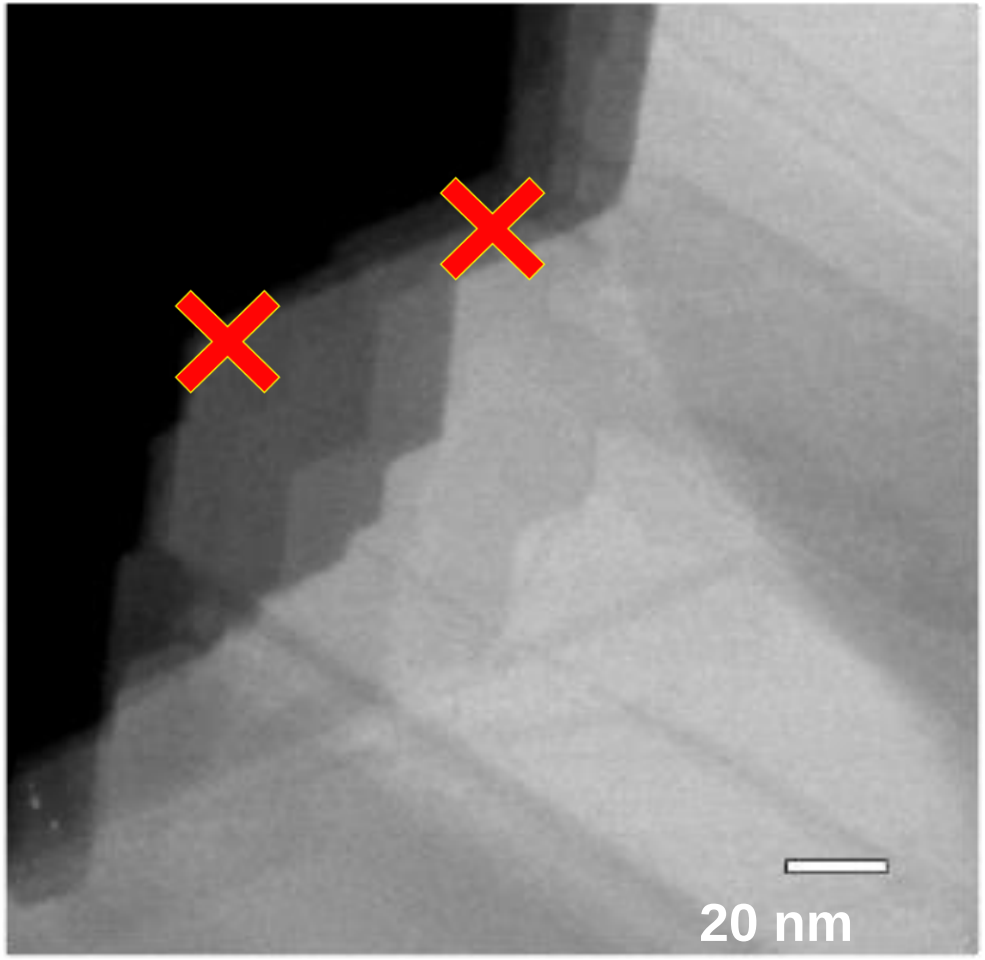


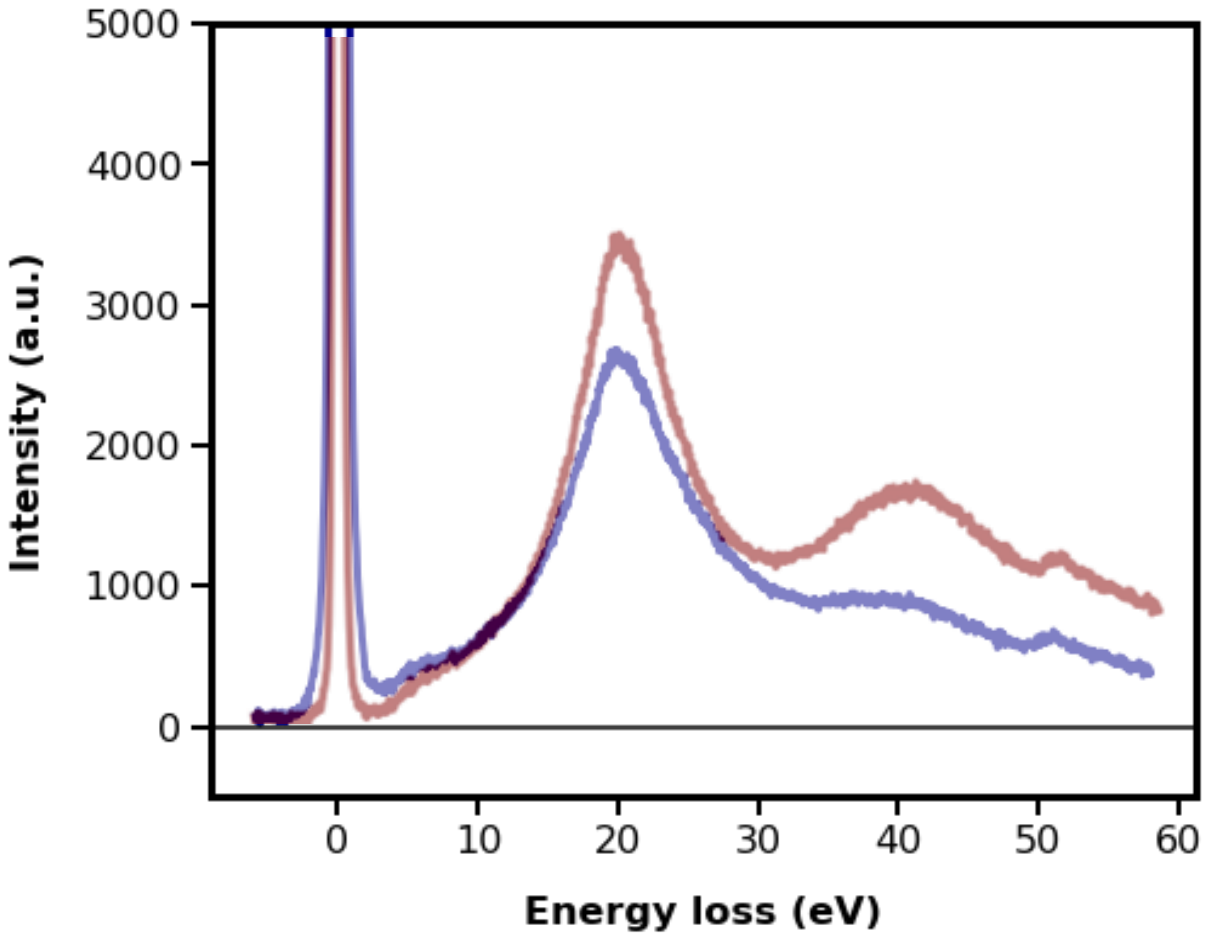

**Key concepts:**

- **Scalarizer:** (any) function that transforms spectrum into measure of interest. Can be integration over interval, parameters of a peak fit, ration of peaks, or more complex analysis

- **Experimental trace:** collection of image patches and associated spectra acquired during experiment. Note that we collect spectra, not only scalarizers

# From Static to Active Learning



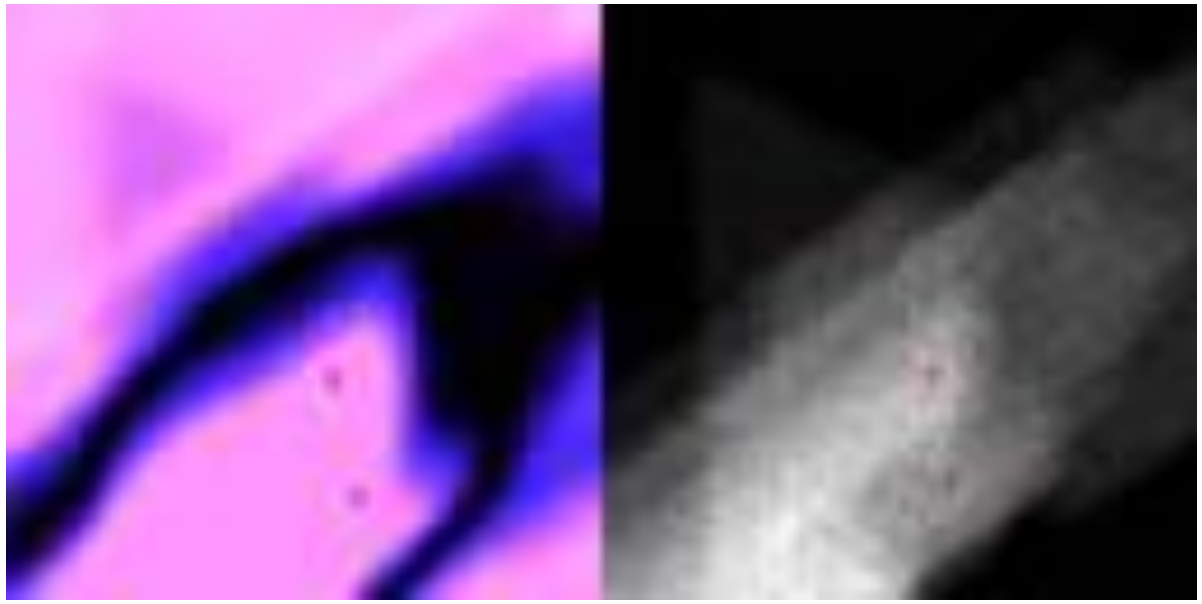

1. What if we have full access to structural information
2. And want to choose locations for (EELS, 4D STEM, CL, EDX) measurements
3. So as to **learn** relationship between structure and spectrum fastest
4. Or **discover** which microstructural elements give rise to specific **desired** spectral features?

# Discovering Regions with Interesting Physics

- **Discovering physics in a "new" material $MnPS_3$**
- **Curve fitting** to help enforce physical processes

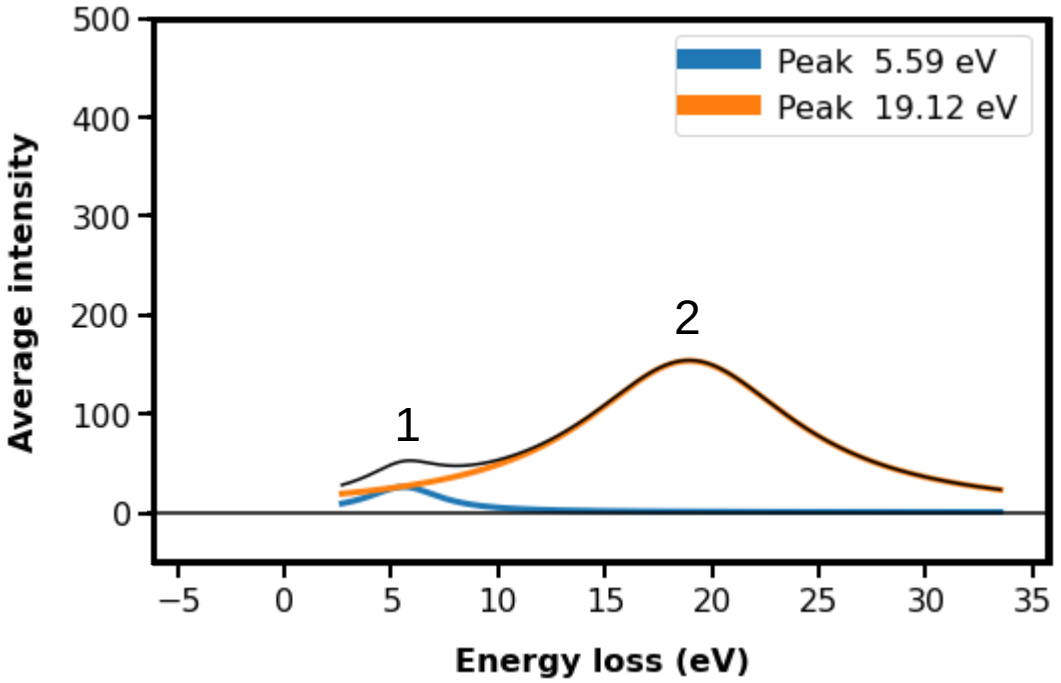
**Acquisition function**

**HAADF-STEM**

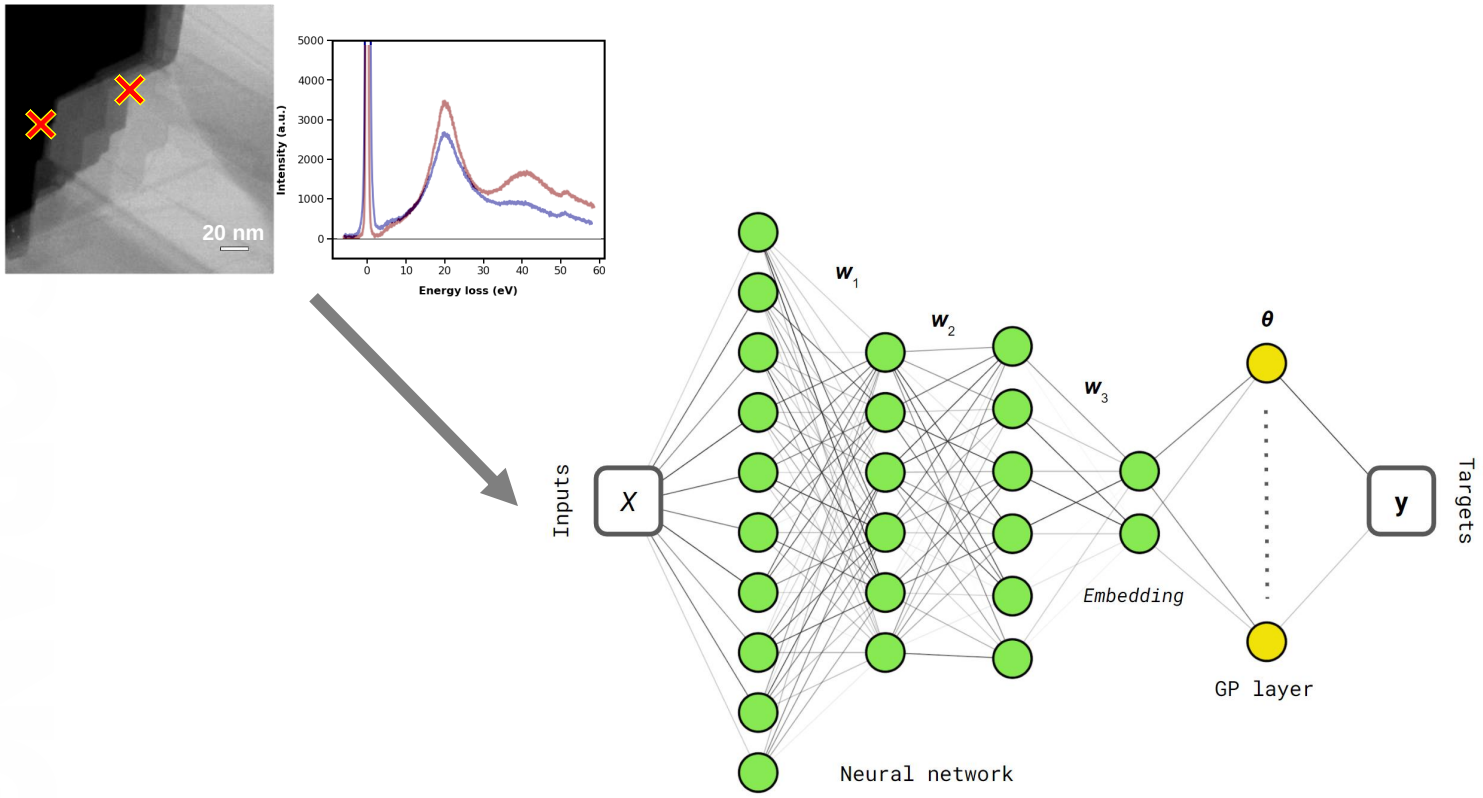


Physics search criteria:

$$Ratio = Peak\ 1 / peak\ 2$$



# Deep Kernel Learning




Specify physics criteria

Acquire **structural data**

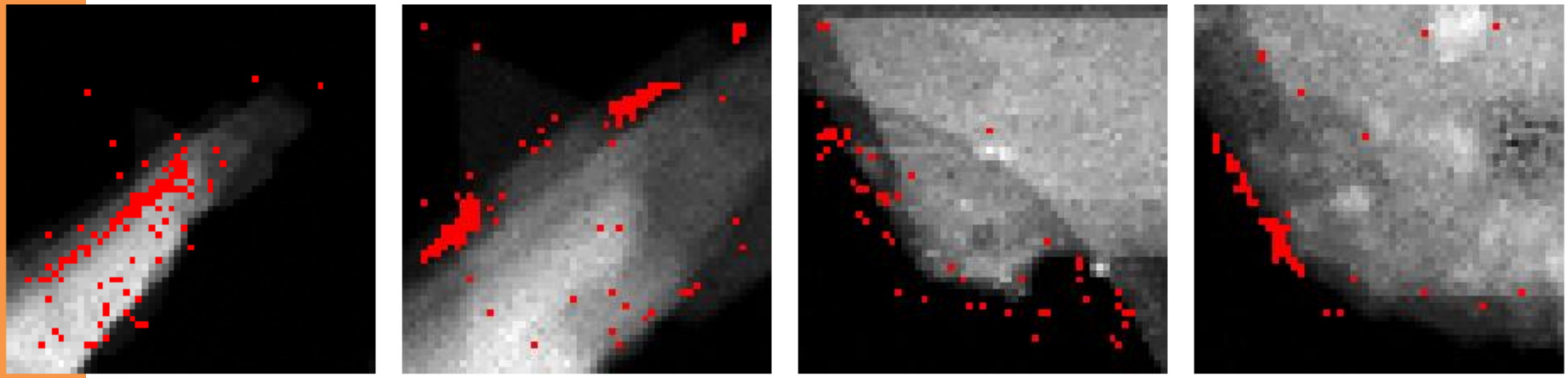
Measure a **spectrum**

Train DKL model with new data

Decide next position (optimize physics criteria)

**Allows navigation of the system to search for physics**

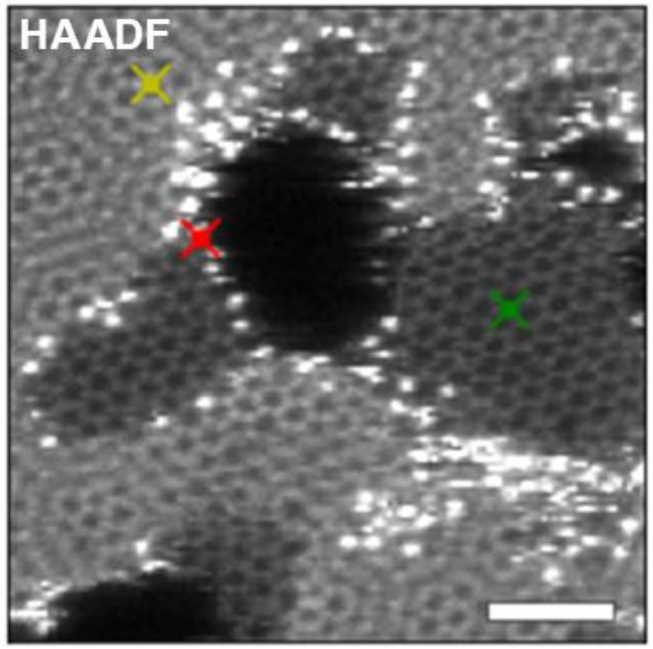
# More Examples of Physics Discovery

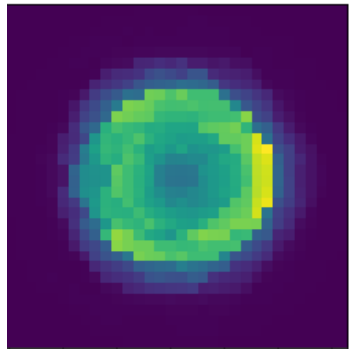


**Discovery pathway depends on the reward structure (scalarizer that defines signature of physics we want to discover)!**

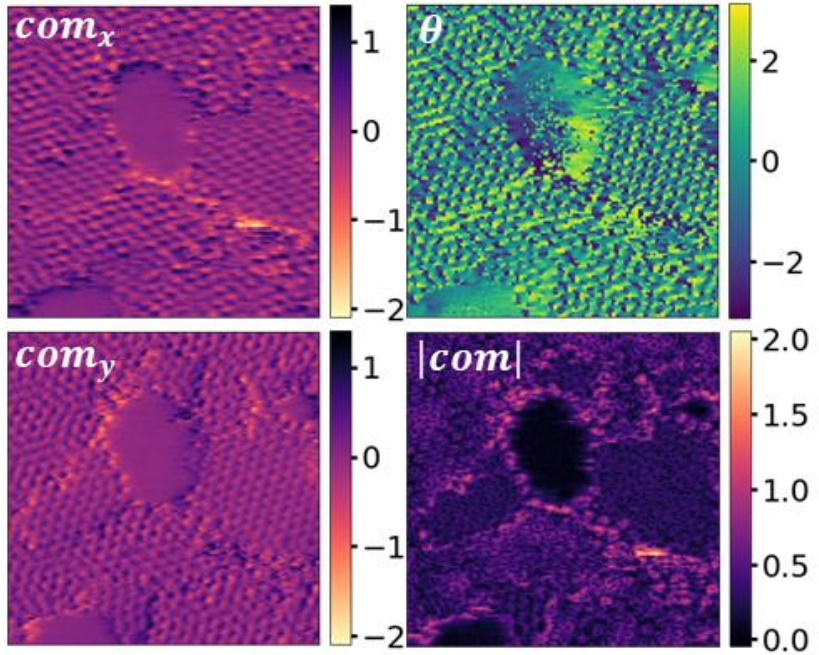
- Currently, we run 4D STEM measurements on a grid.
- What if we want to explore smarter workflows – where microscope chooses where to take 4D STEM measurements?
- **Direct:** We can do it for a priori known objects of interest
- **Inverse:** Or we can aim to discover objects which have predefined signatures of interest in 4D STEM data

# 4D STEM: Grid, Direct, and Inverse



Quantities to explore
- Electric field
- Potential
- Charge density
- Strain

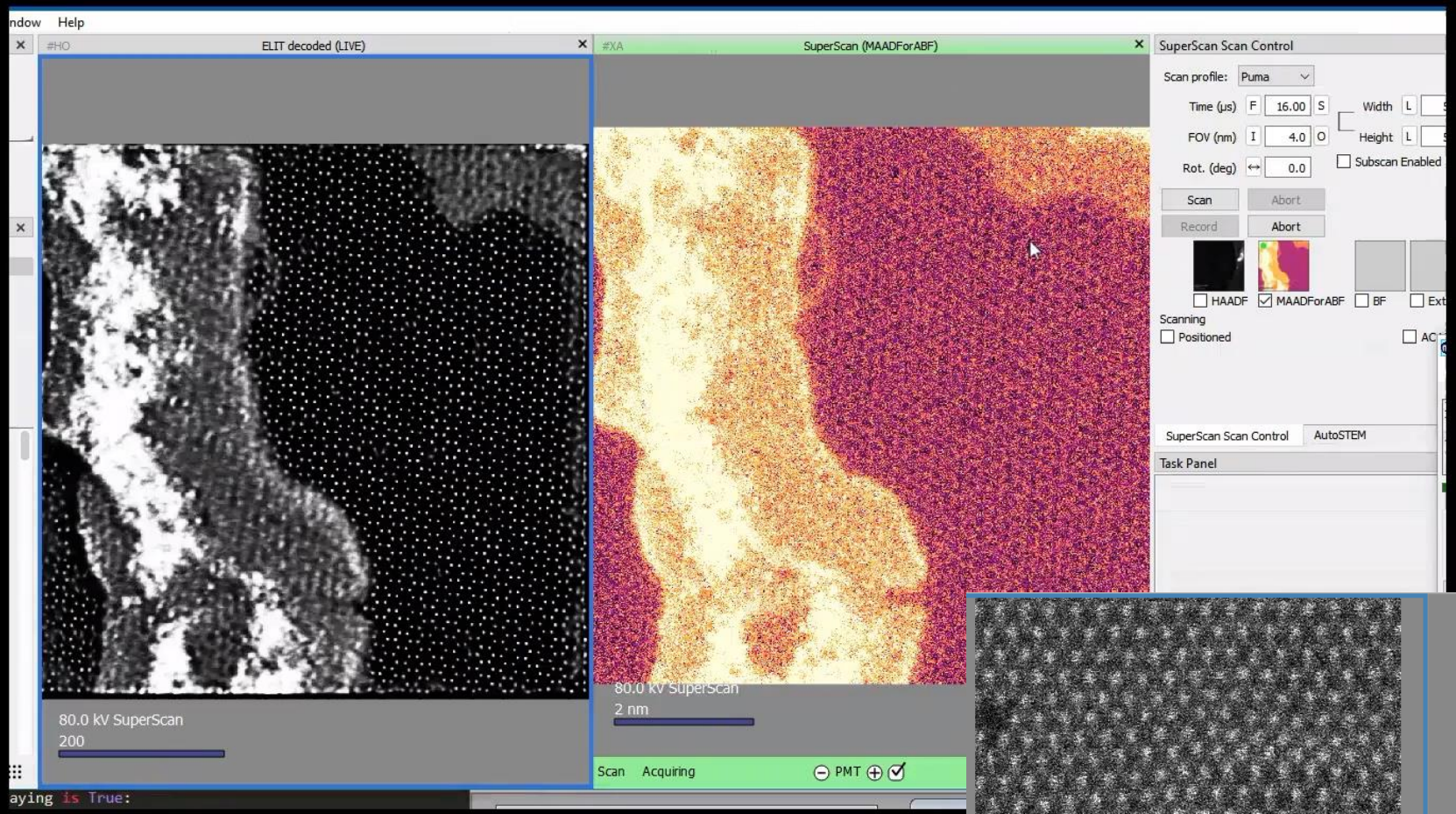



- What can we say about structure?
- Interesting functionalities are expected at the certain structural elements
- We can guess some; we have to discover others
- Multiple goals while running experiment

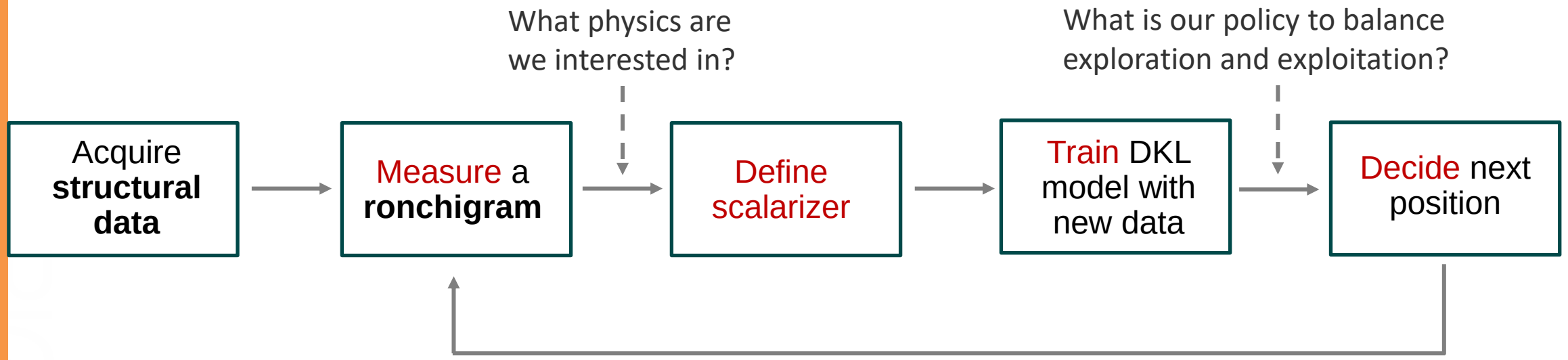
- **Policy:** **what do we do depending on observation**
- **Reward:** **what do we hope to achieve**
- **Value:** **anticipated reward**

# Direct experiment: ELIT (2021)

Implementation: Kevin Roccapriore, Ayana Ghosh, Sergei V. Kalinin & Maxim Ziatdinov



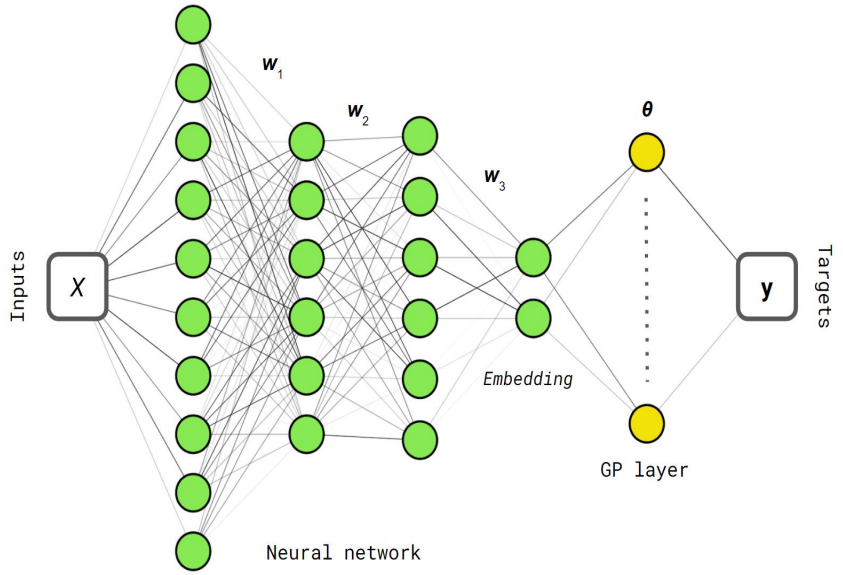
# Inverse: Deep Kernel Learning based BO



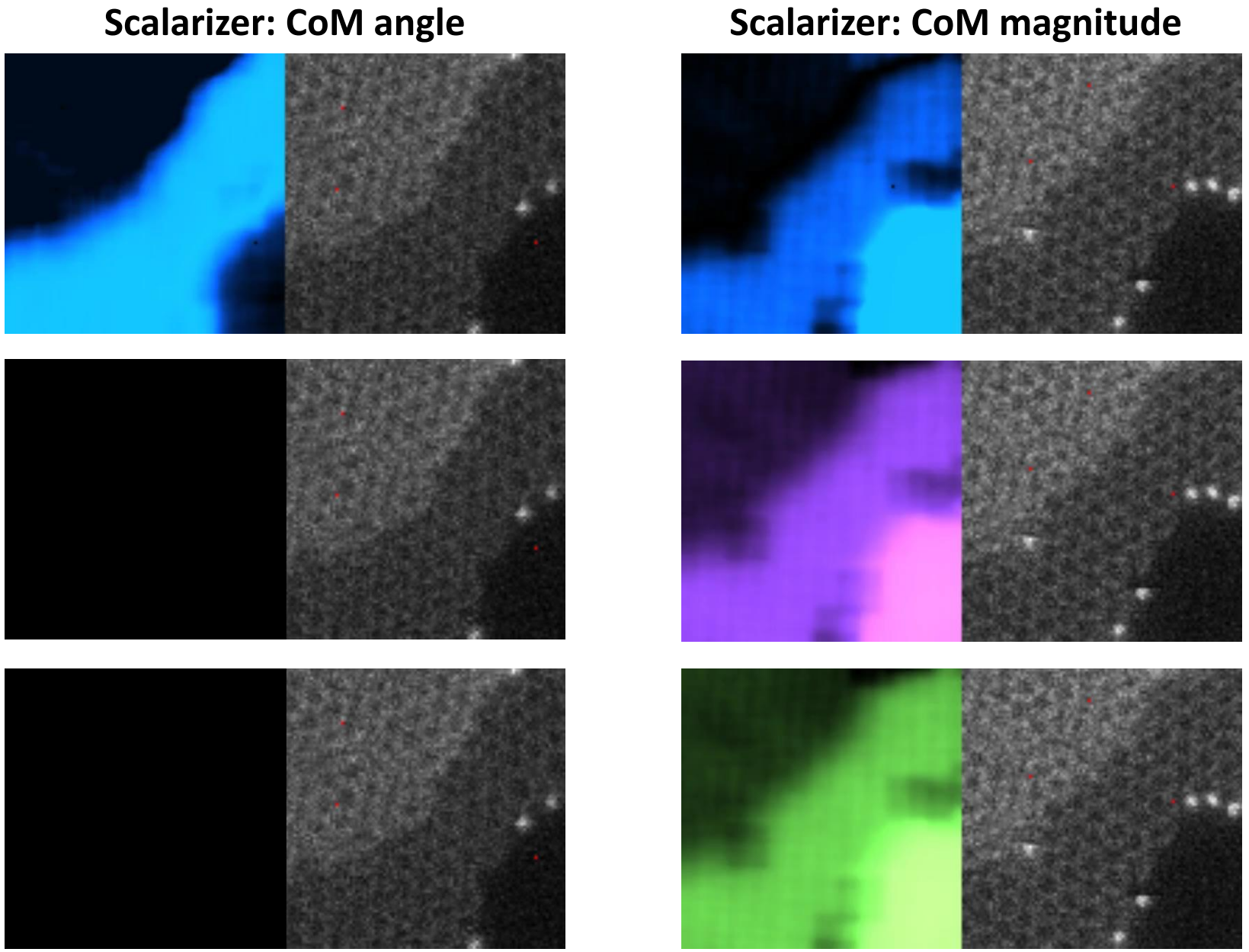

**Key concepts:**

- **Scalarizer:** (any) function that transforms spectrum into measure of interest. Can be integration over interval, parameters of a peak fit, ration of peaks, or more complex analysis

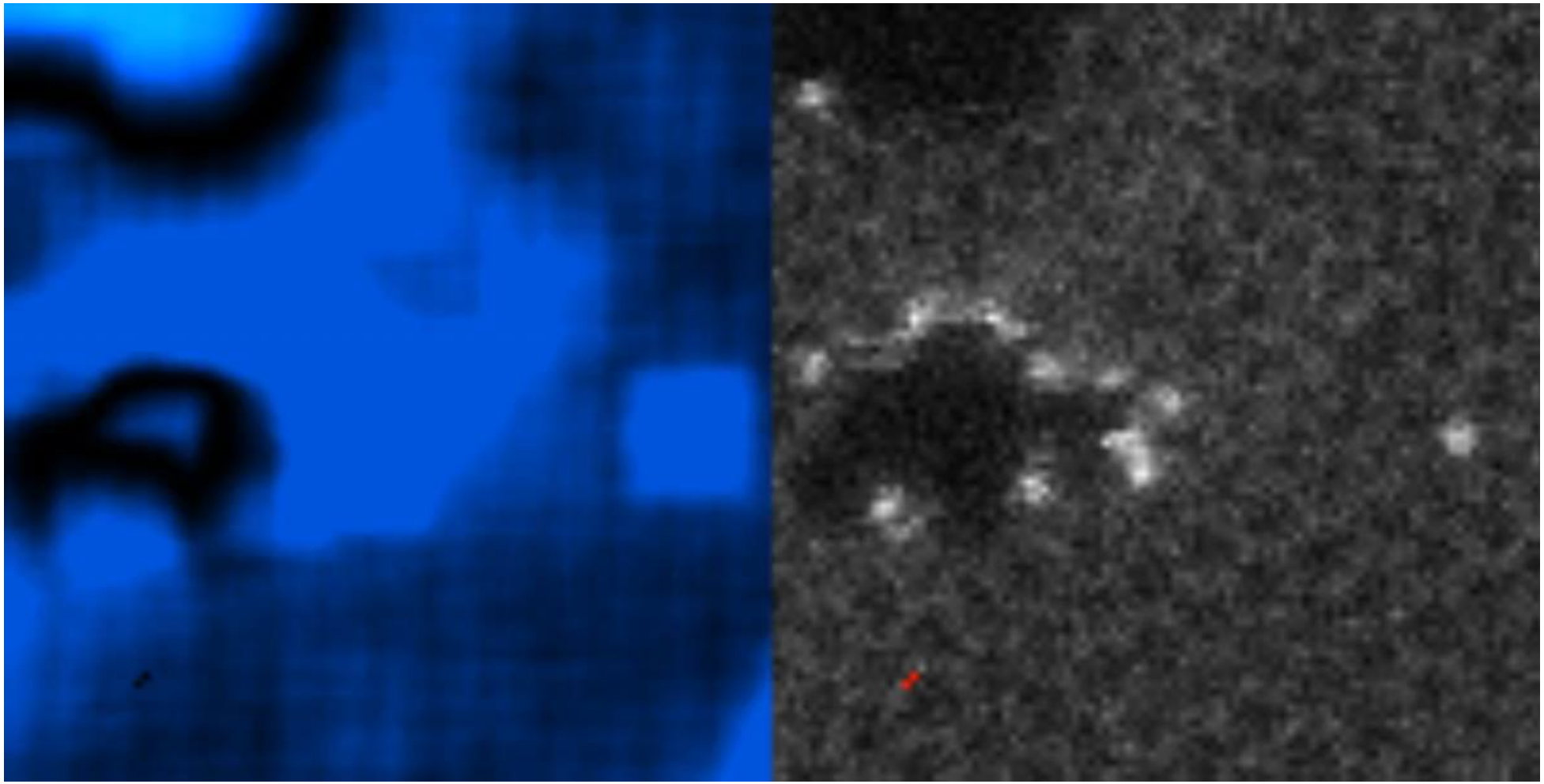
- **Experimental trace:** collection of image patches and associated spectra acquired during experiment. Note that we collect spectra, not only scalarizers



# DKL on pre-acquired data



# DKL on Active Microscope

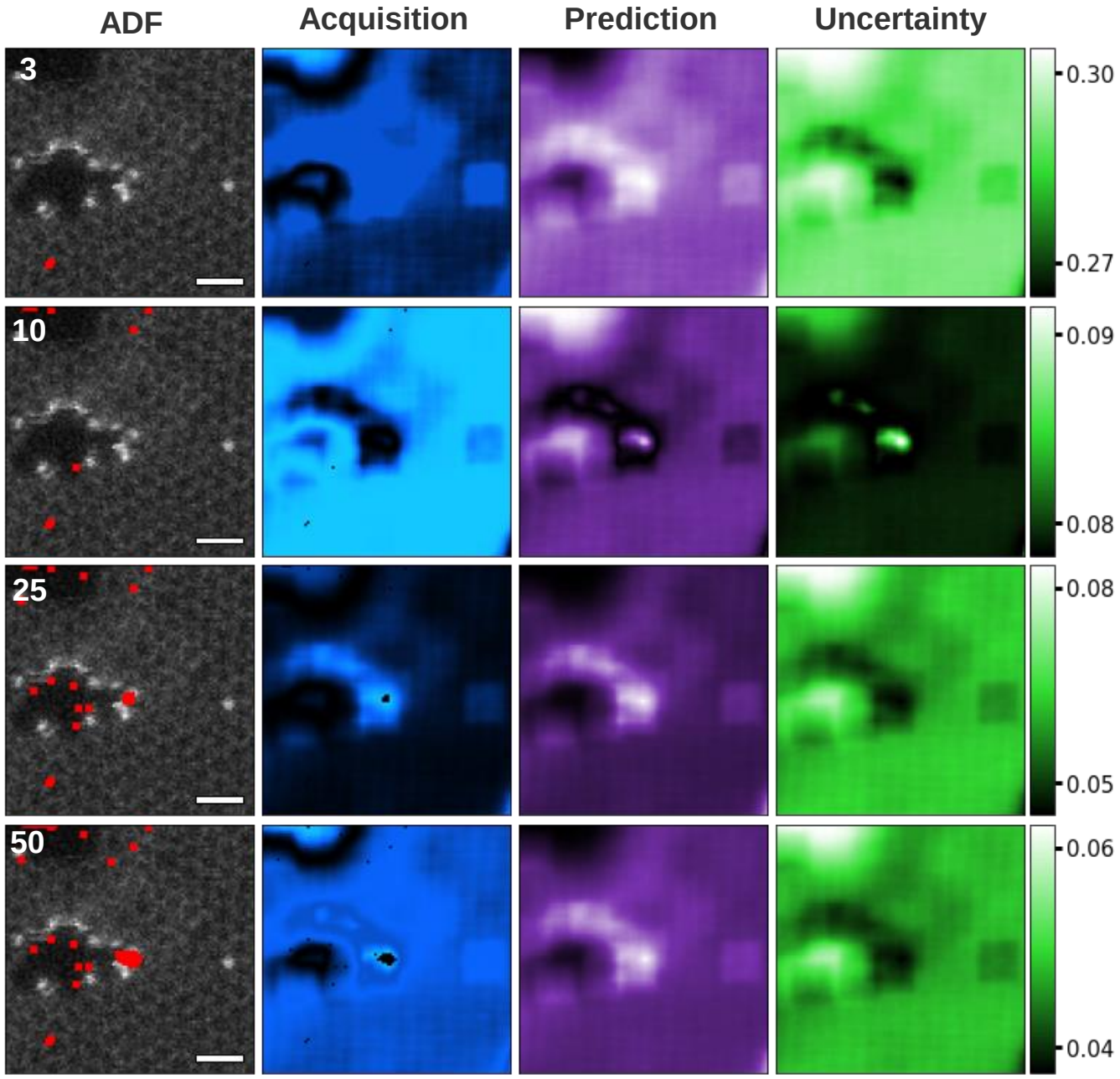


- Different **acquisition functions** can be used:
  - Expected Improvement (**EI**) (usually what was used)
  - Upper Confidence Bound (**UCB**), etc
- Usually based on some combination of **prediction** and **uncertainty**.

# A closer look

## Scalarizer: *CoM* magnitude

- High uncertainty @ start, but fairly quickly reduces
- Prediction actually doesn't drastically change throughout experiment
    - Structure-property relationship here is fairly rapidly learned
- Note the training can be halted after some criterion is met, making remainder of experiment go much quicker



# Does it always work?