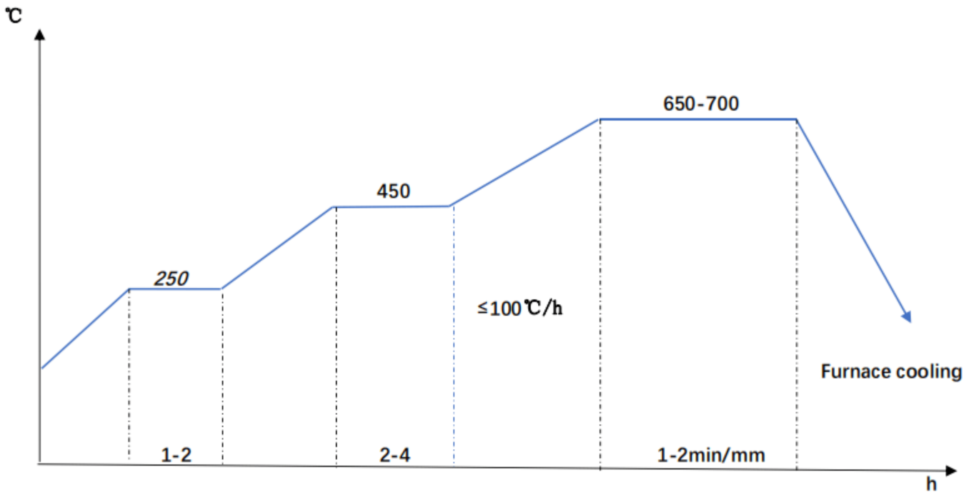
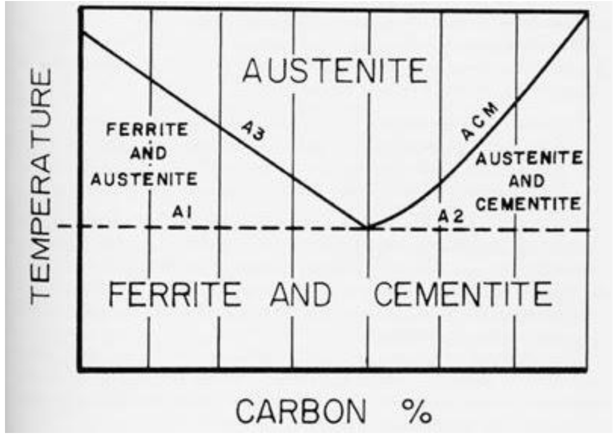
THE UNIVERSITY of TENNESSEE KNOXVILLE

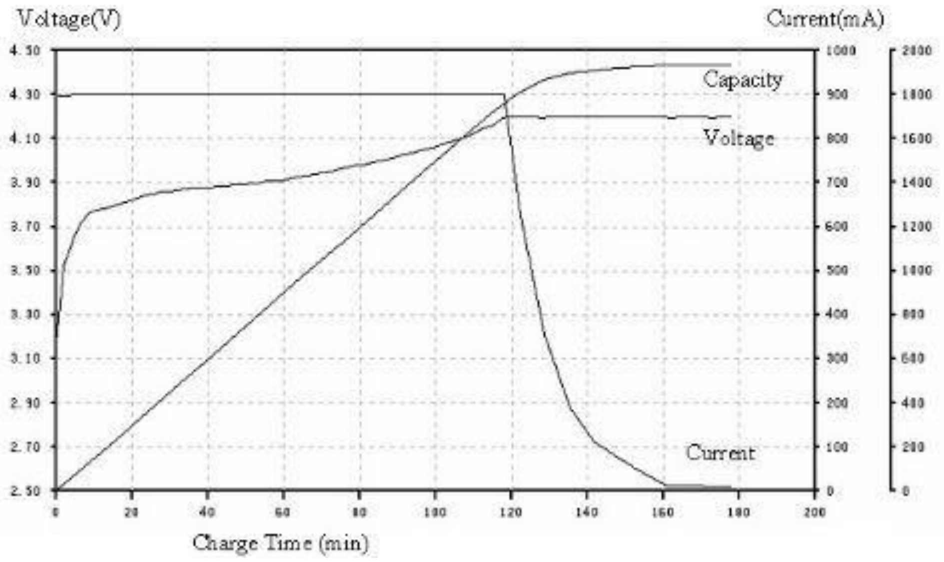
Deep Kernel Learning – II:
Process Optimization

Sergei V. Kalinin

Making materials: process trajectories



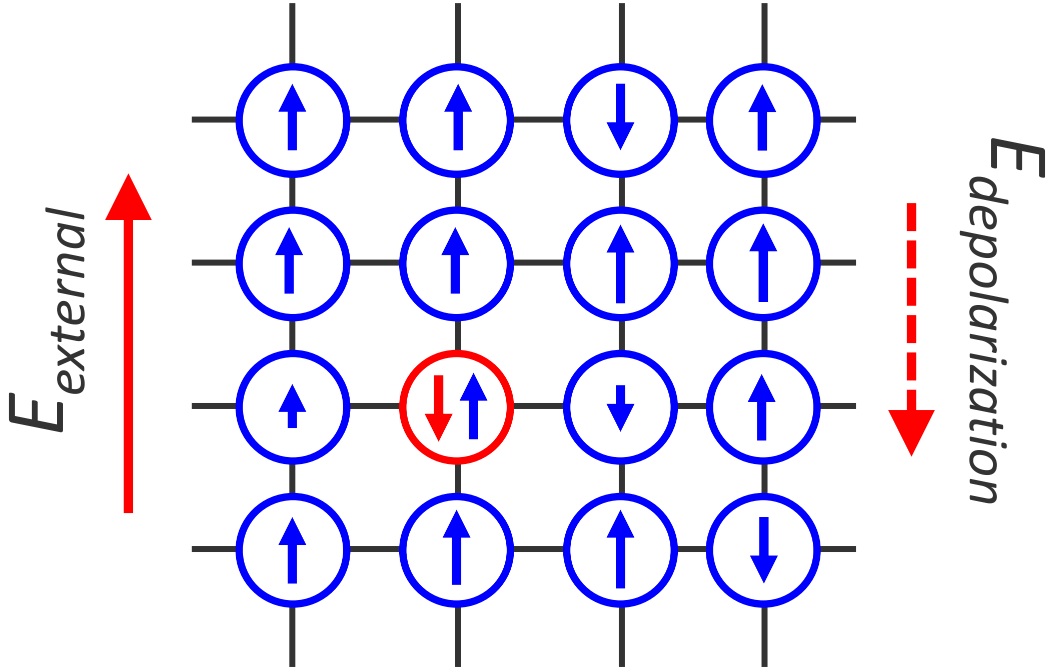


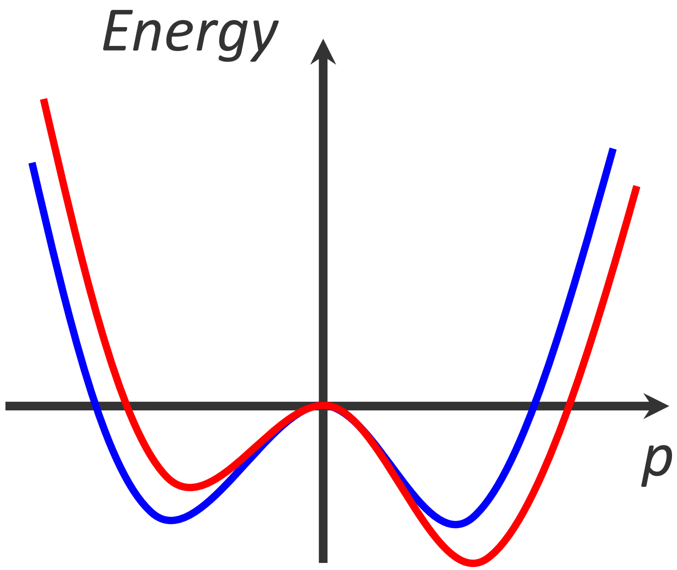


- Making steel: complicated and took a lot of time optimize
- Charging battery: obvious economic impact
- Manufacturing: Annealing hybrid perovskite thin films
- Poling ferroelectric

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?

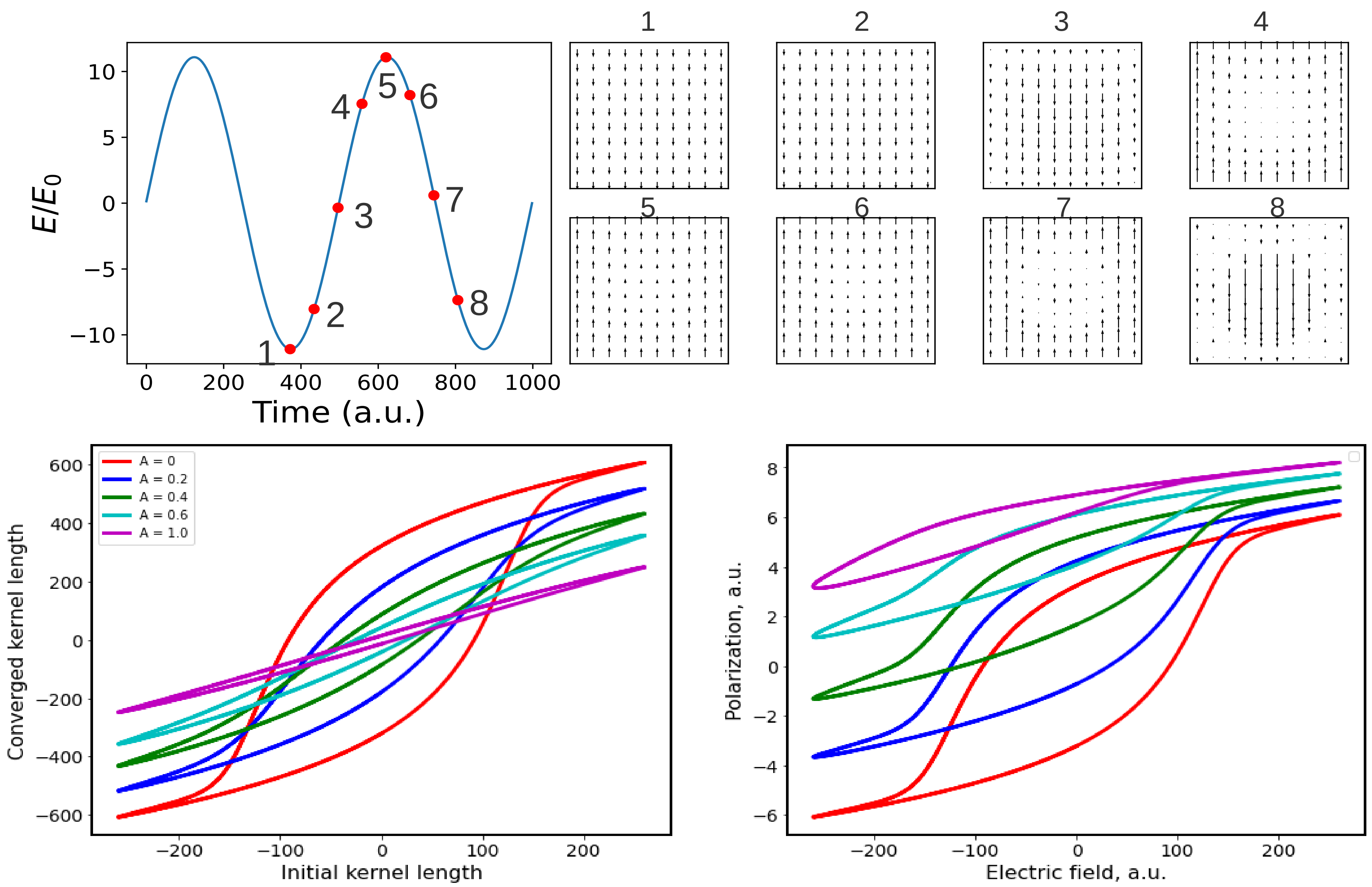
FerroSIM: the simplest interesting ferroelectric





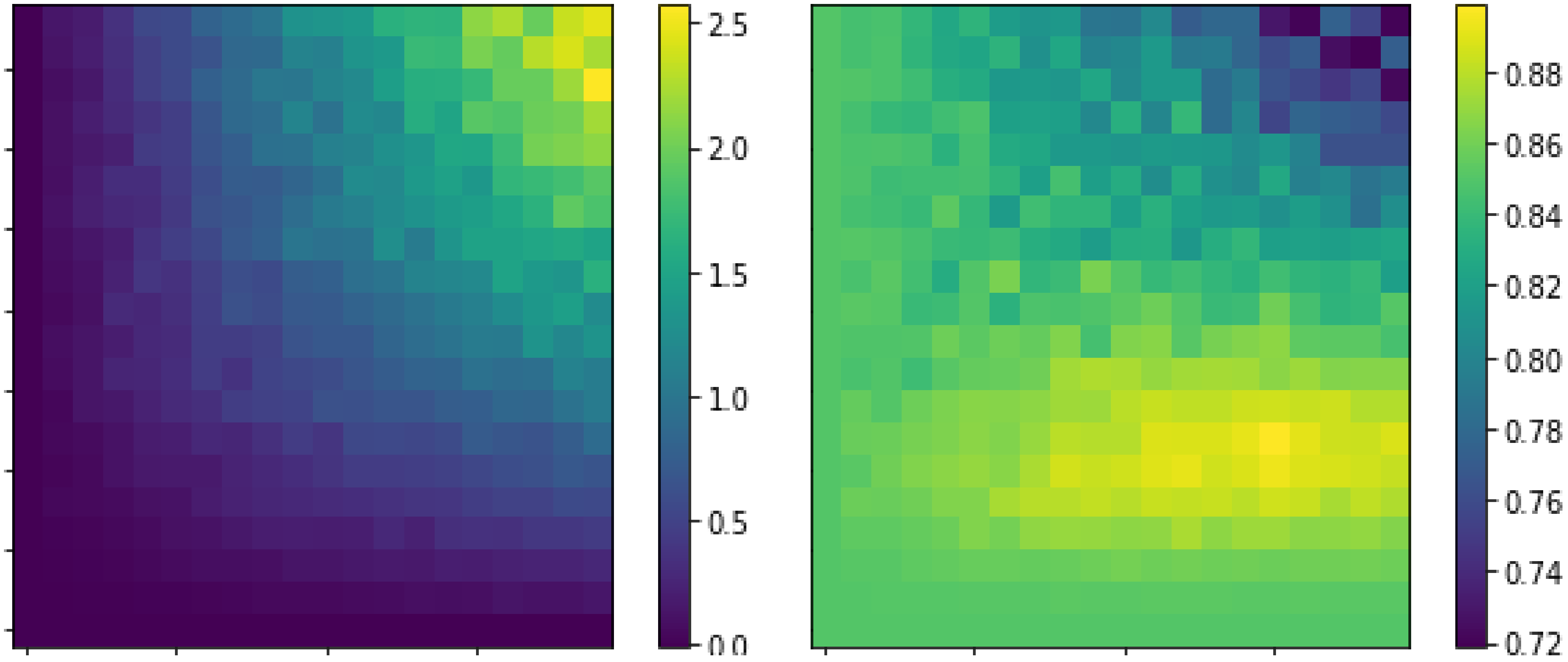
- A discrete square lattice where a continuous polarization vector resides at each lattice site
- The local free energy at each site takes the GLD form:
  - $F_{ij} = \alpha_1 \left(p_{x_{ij}}^2 + p_{y_{ij}}^2\right) + \alpha_2 \left(p_{x_{ij}}^4 + p_{y_{ij}}^4\right) + \alpha_3 p_{x_{ij}}^2 p_{y_{ij}}^2 - E_{loc_{x_{ij}}} p_{x_{ij}} - E_{loc_{y_{ij}}} p_{y_{ij}}$
  - 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 $\frac{\gamma d p_{i,j}}{dt} = -\frac{\partial F}{\partial p_{i,j}}$

Microstates and Macroscopic Observables



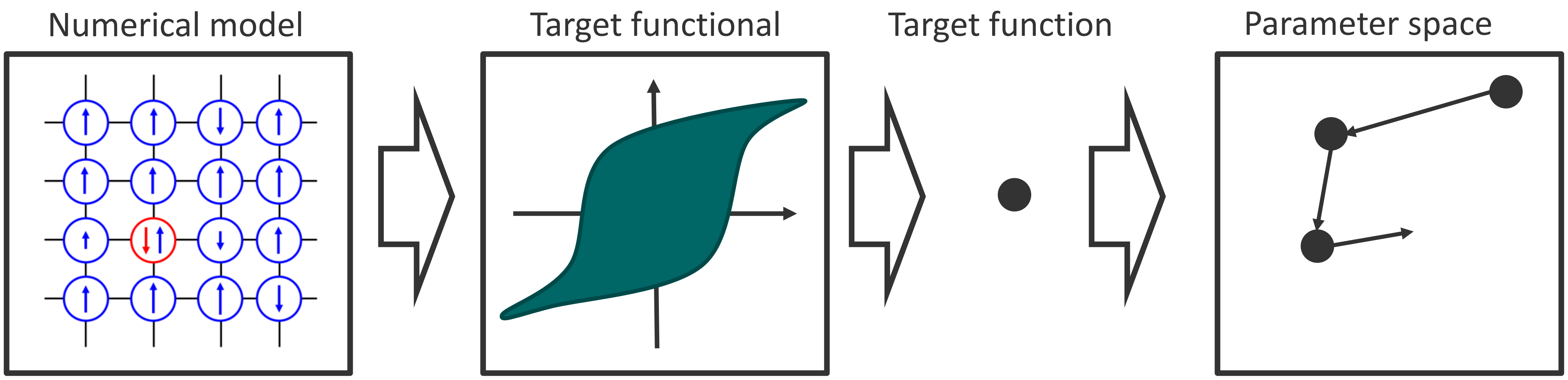
Global response vs. model parameters

Parameter space 1: Hamiltonian
Parameter space 2: Field history



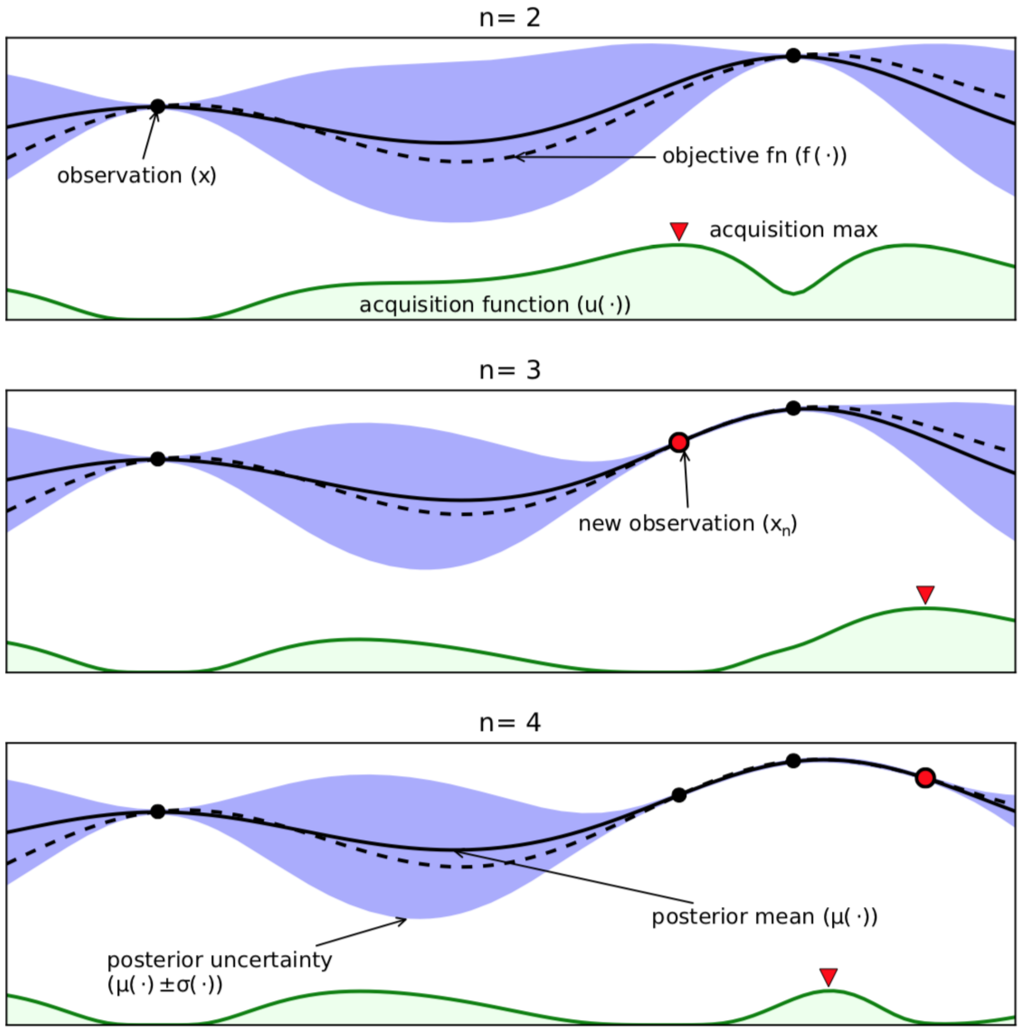
For small dimensional parameter spaces, we can evaluate global responses via the grid search

Can we do better then grid search?



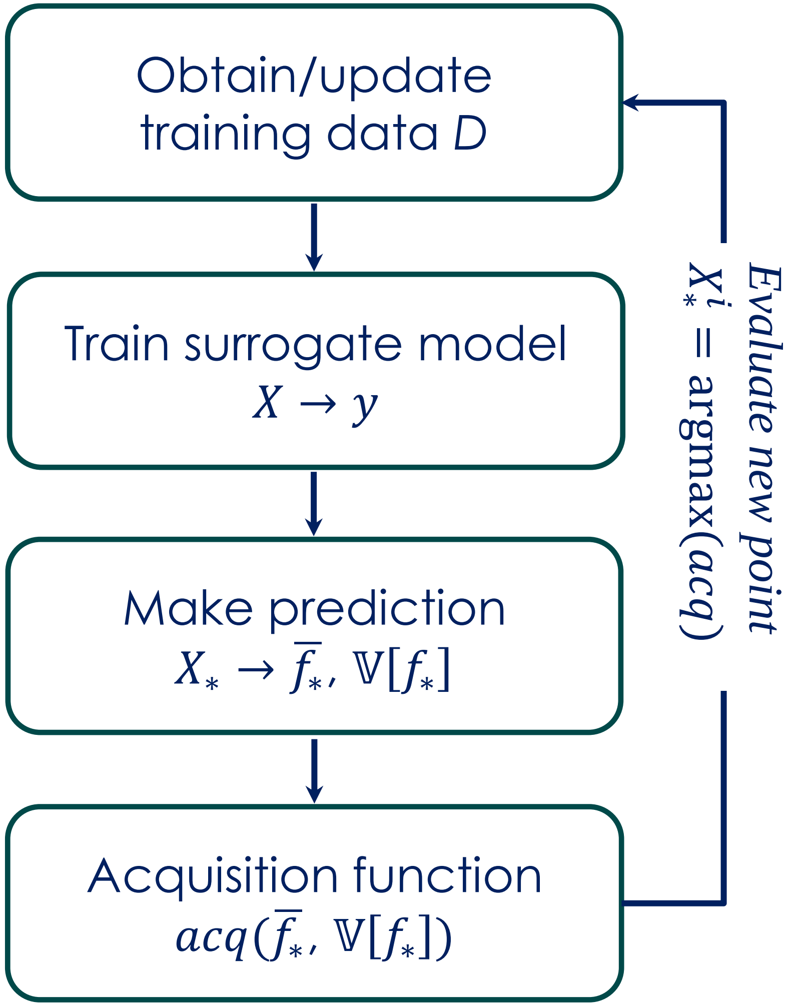
Or we can use simple Gaussian Process-based Bayesian Optimization to do so

Bayesian Optimization!

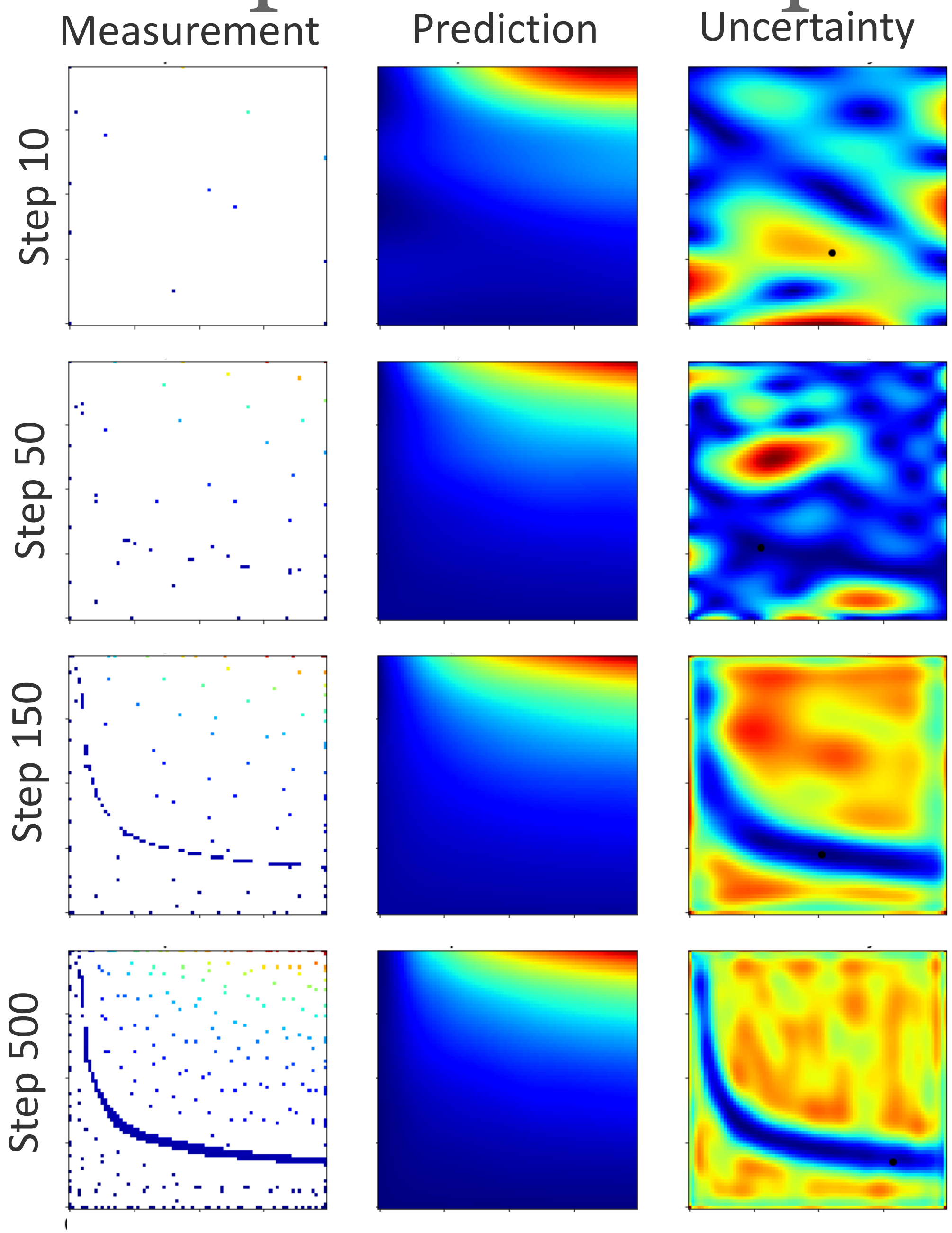


N. de Freitas et al., Taking the Human Out of the Loop: A Review of Bayesian Optimization , *Proceedings of the IEEE* **104**, 148 (2015)

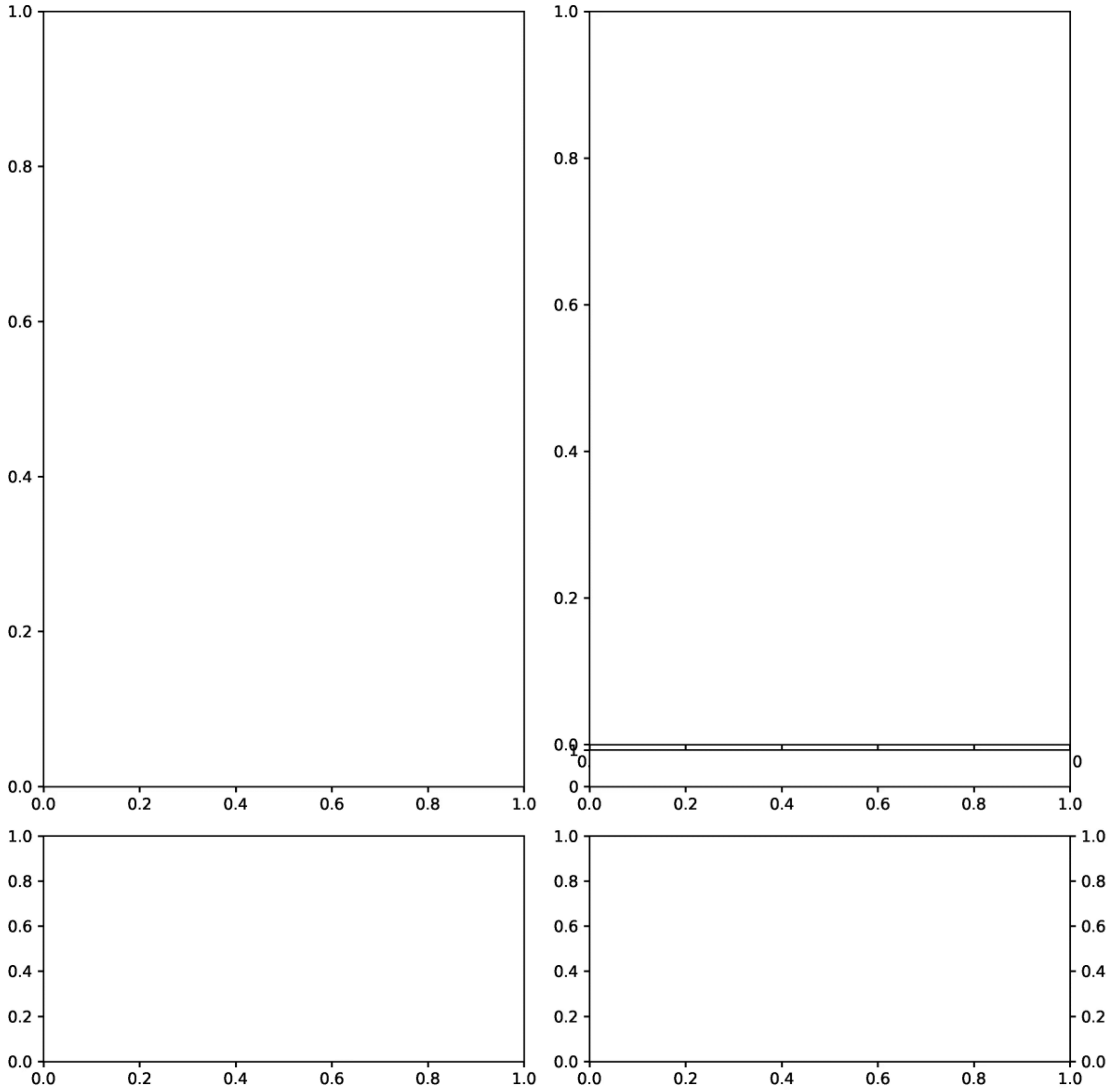
X, y: (sparse) Training data
X_*: New (not yet evaluated) points



BO exploration of parameter space



But what about trajectories?



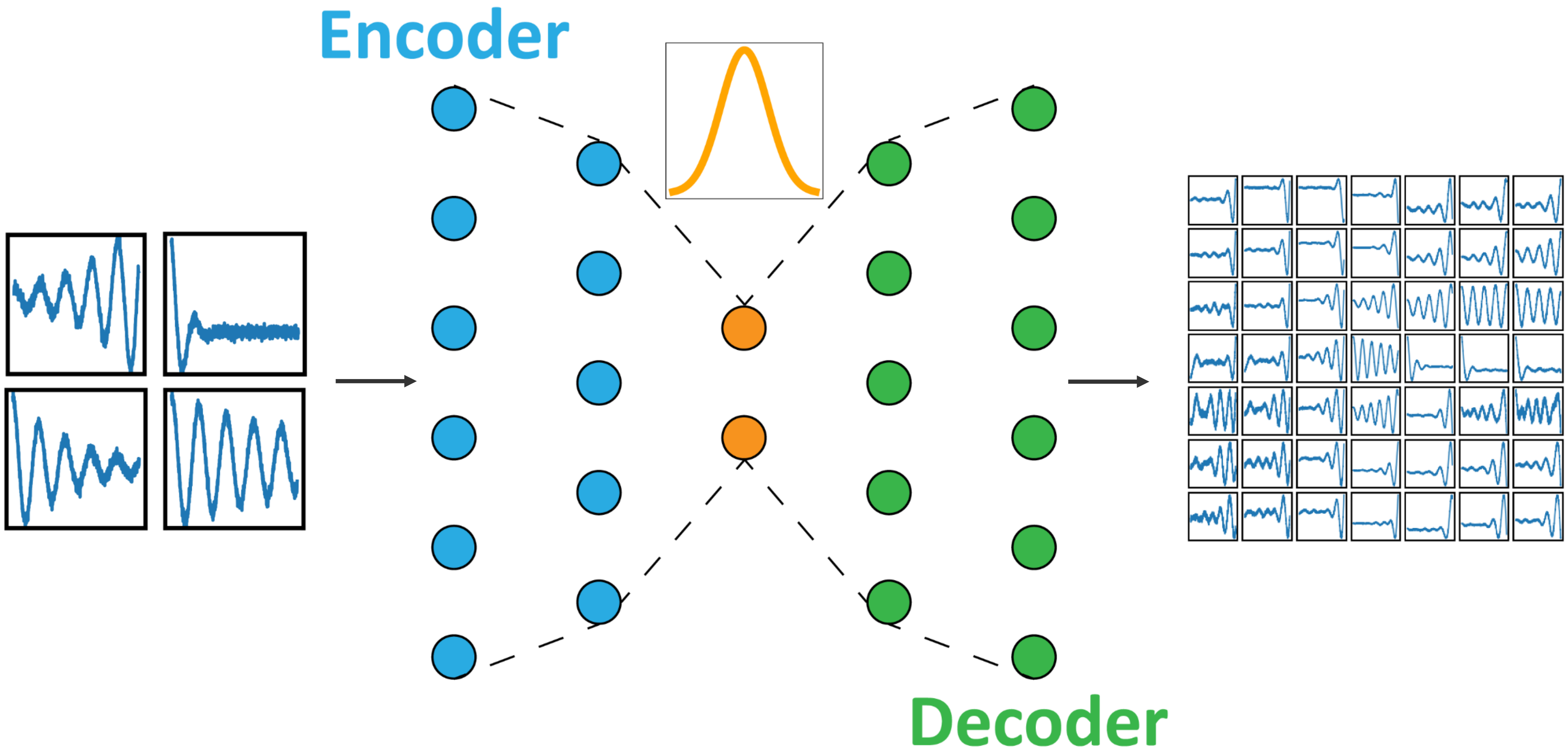
- The model has large number of microstates
- The global state depends on history, i.e. dependence of field vs. time
- Can we somehow optimize the chosen global state in the space of possible histories?

- 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?

Can VAE help?

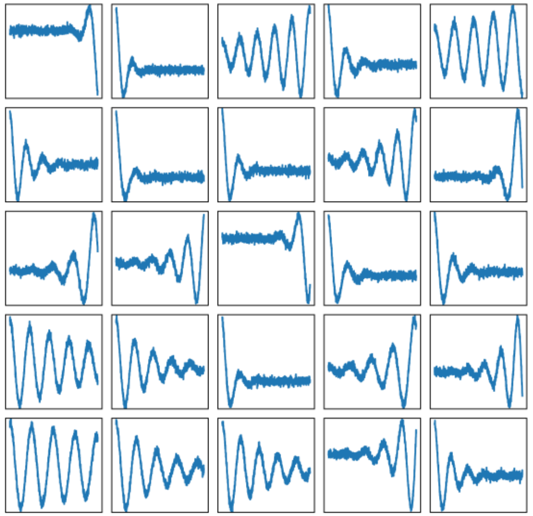


VAE encoding of domain trajectories

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

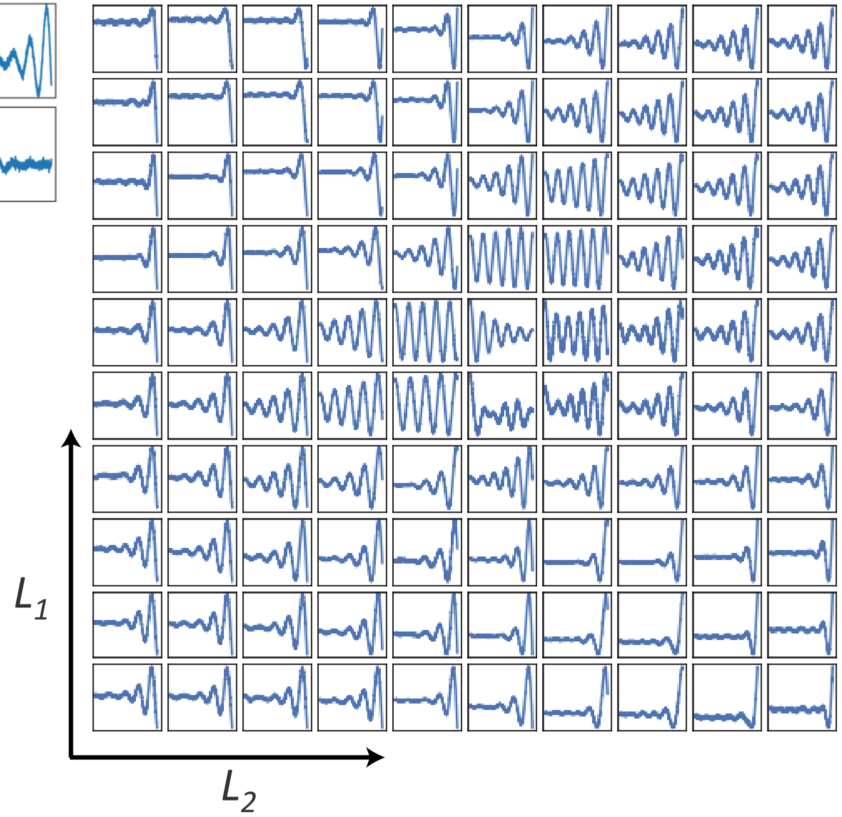
- These electric fields are divided into 900 discrete time steps.

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

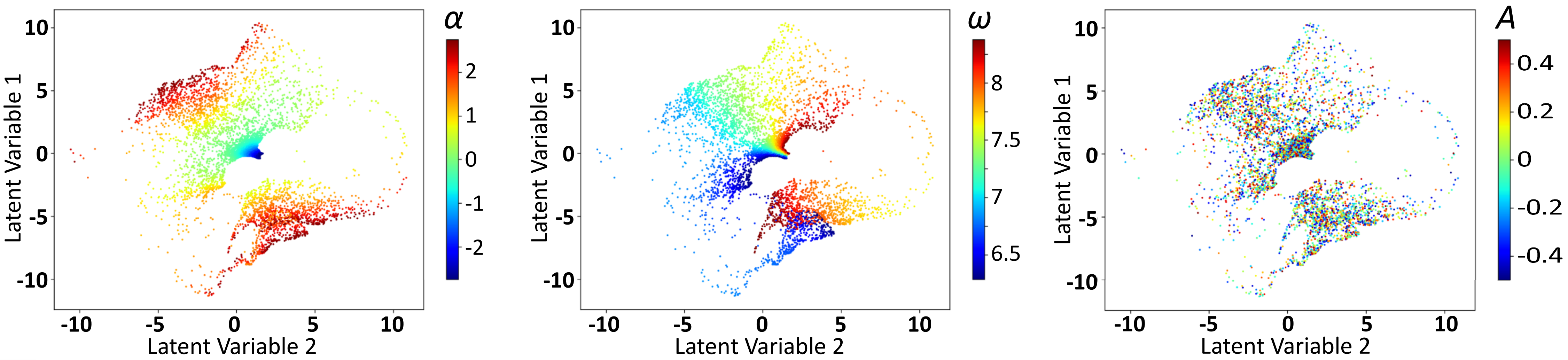


Example Trajectories

Decoded Latent Space

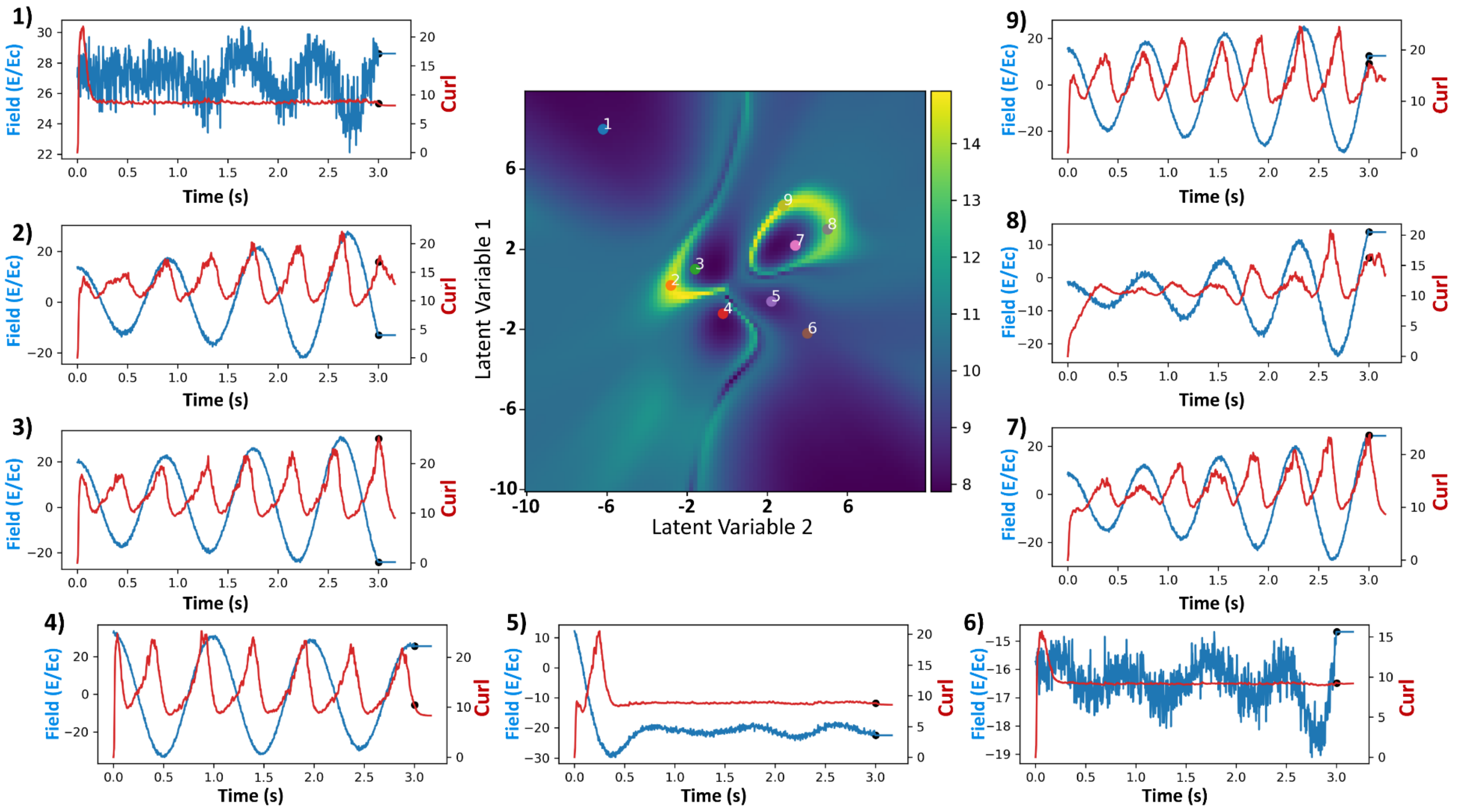


Latent space distributions



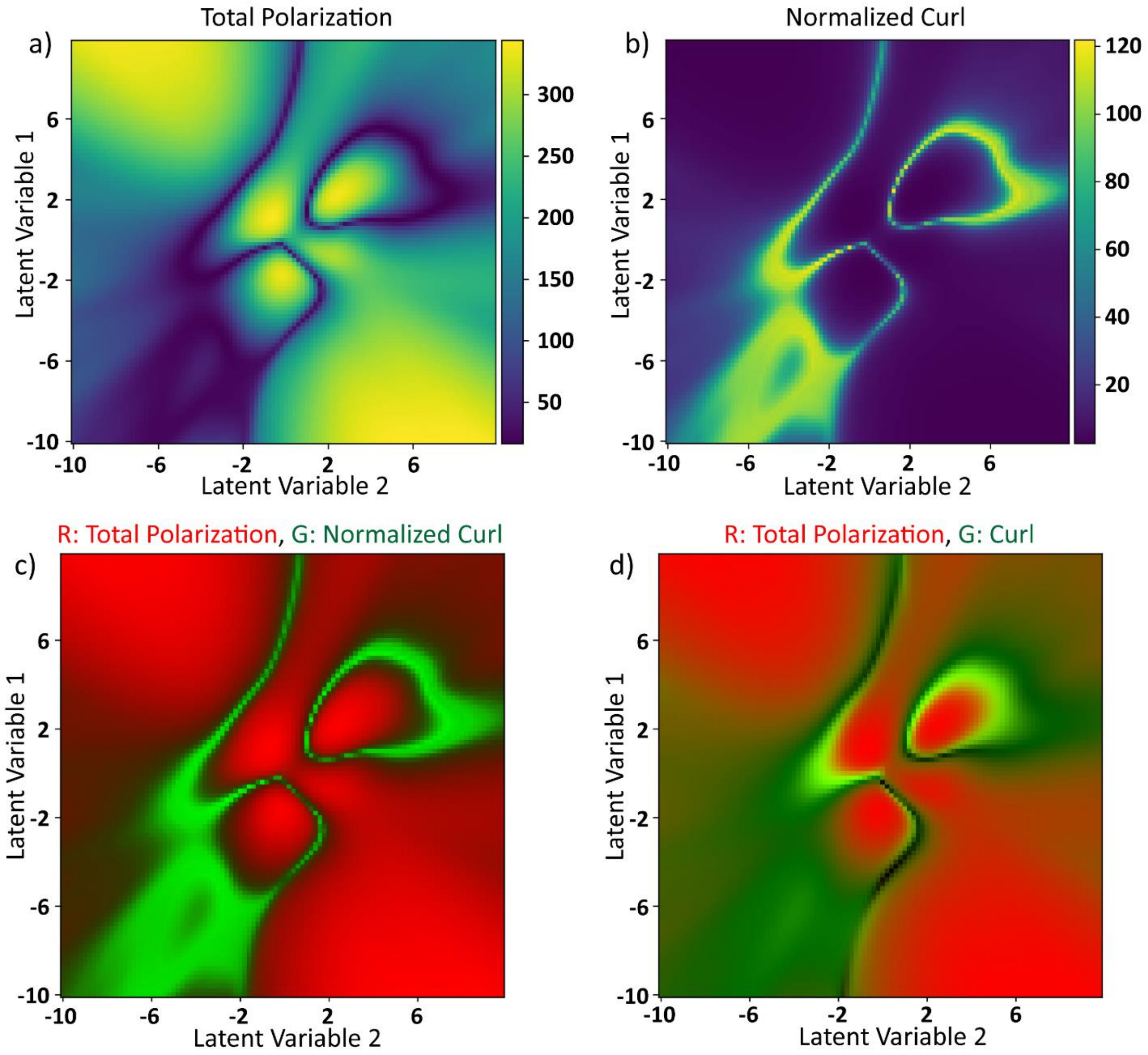
Ground truth target function

- 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.

Exploring the curl surface



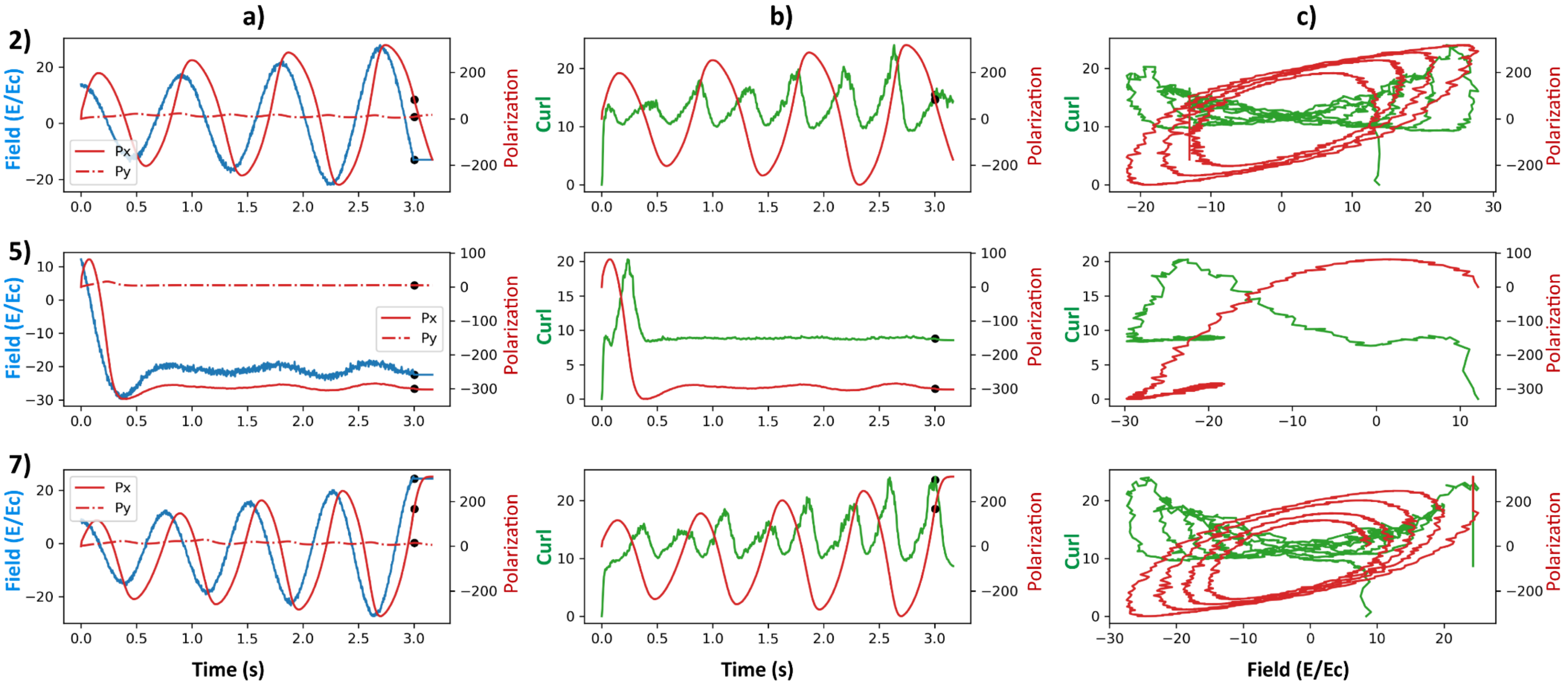
Normalized Curl

- At the end of the simulations, the polarization vector at each lattice site is normalized
- These normalized polarization vectors are then used to recalculate curl
- We will refer to it as the normalized curl
- It is supposed to estimate how much the vector field rotates without considering the magnitude of the field

Observations

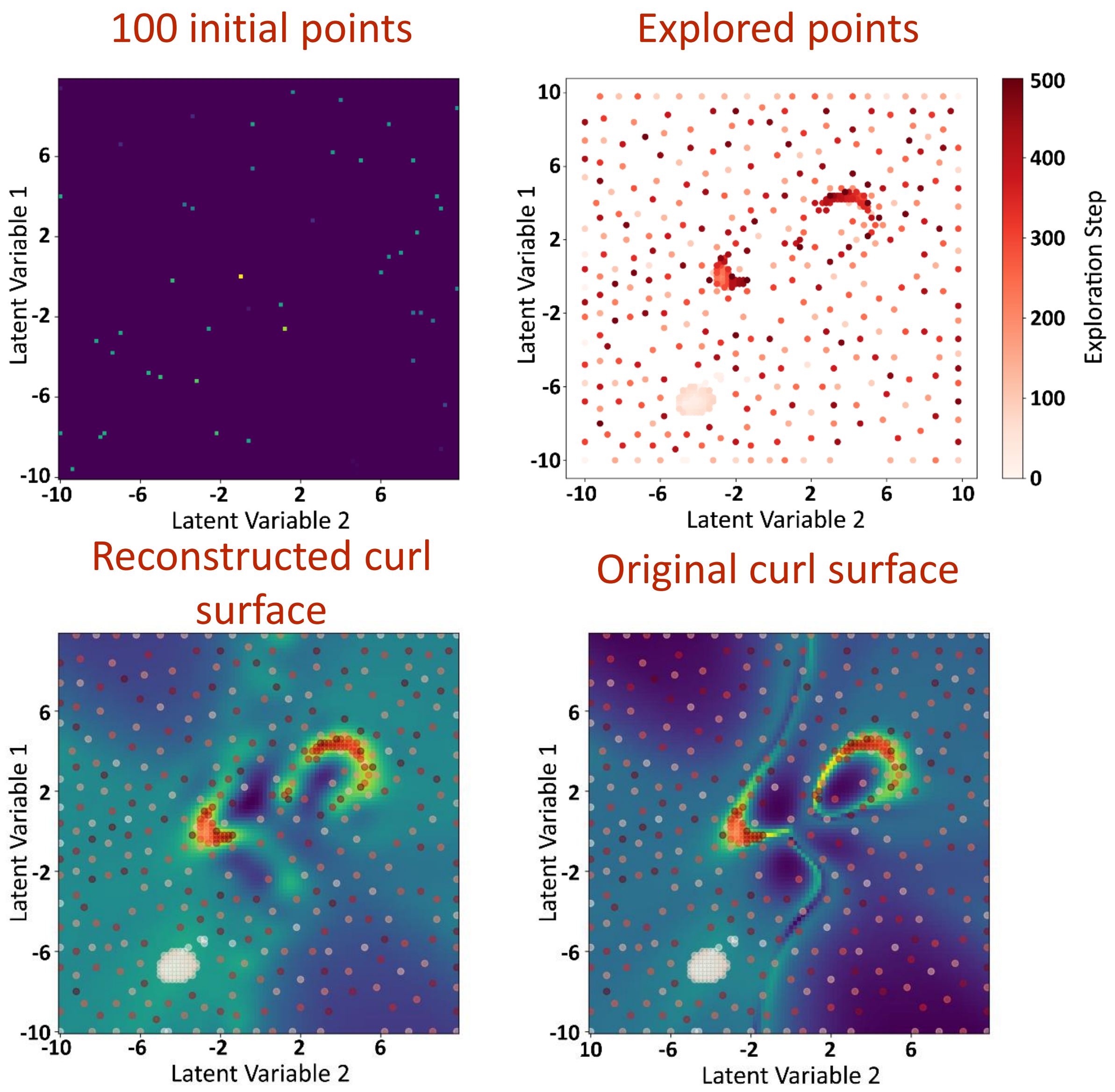
- Normalized curl is inversely proportional to the magnitude of polarization
- The system is allowed to be in the most chaotic state when the polarization is the least as the effect of coupling terms is low
- The system's polarization is at the lowest the coercive field

Exploring the curl surface



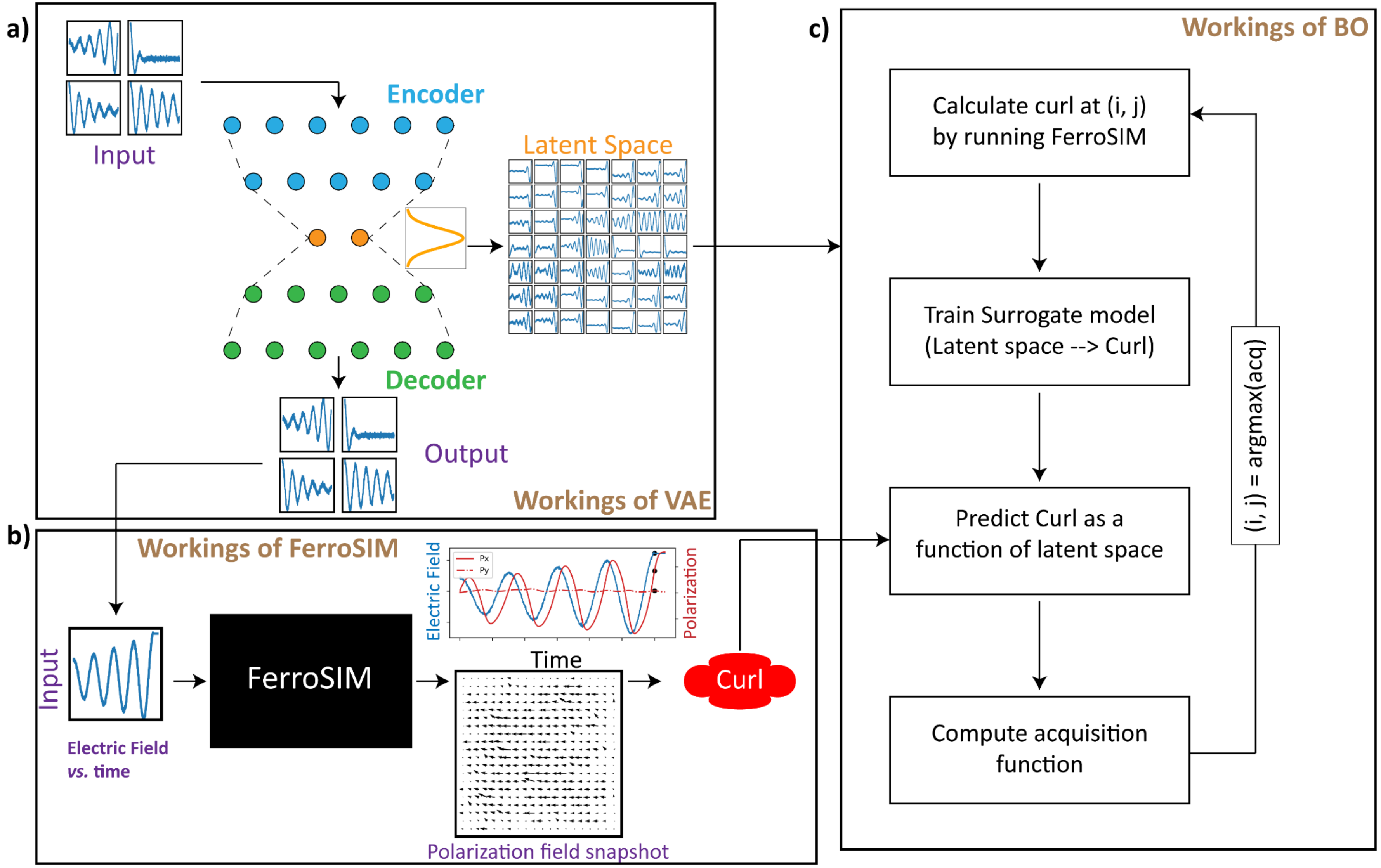
- The system’s polarization is at the lowest the coercive field (A state of maximum normalized curl)
- But the curl is also a function of magnitude of the polarization
- Hence, the magnitude of the curl is maximum a few steps after the coercive field where the polarization grows in magnitude just enough that the coupling terms do not take over to kill the curl in the system
- This time coincides with the time it takes the electric field to reach the maximum from the coercive field, hence the overlap of the local maxima of curl and electric field

Bayesian Optimization in the Latent Space

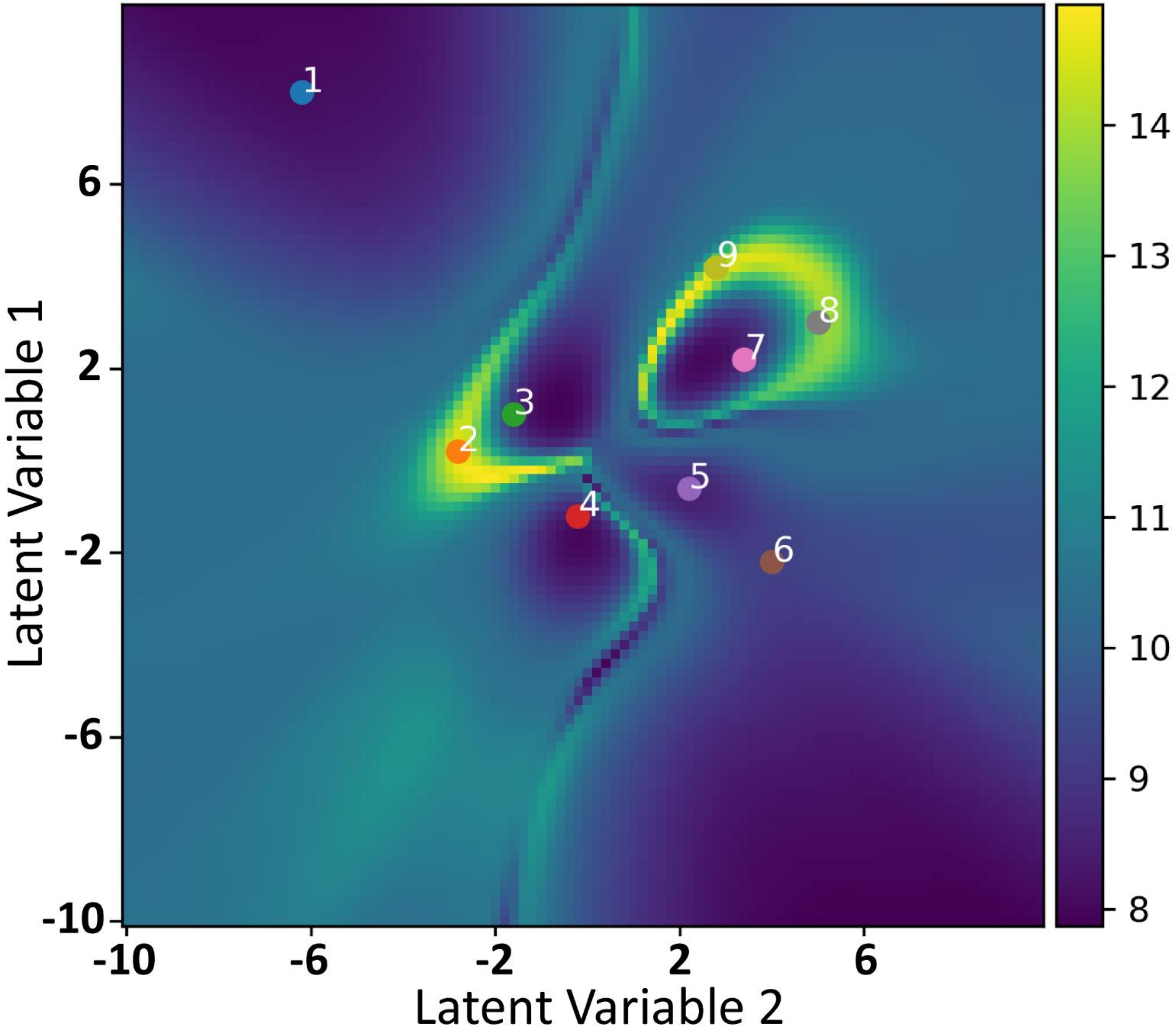


- 100 initialization points and the BO explored the latent space for the next 500 points
- Acq function: μ + 10σ
- So, at the end BO only explored a total of 600 points out of 10,000 points the latent space is divided into
- Caveat: we had to tune the Acq with the ground truth data known

Putting everything together



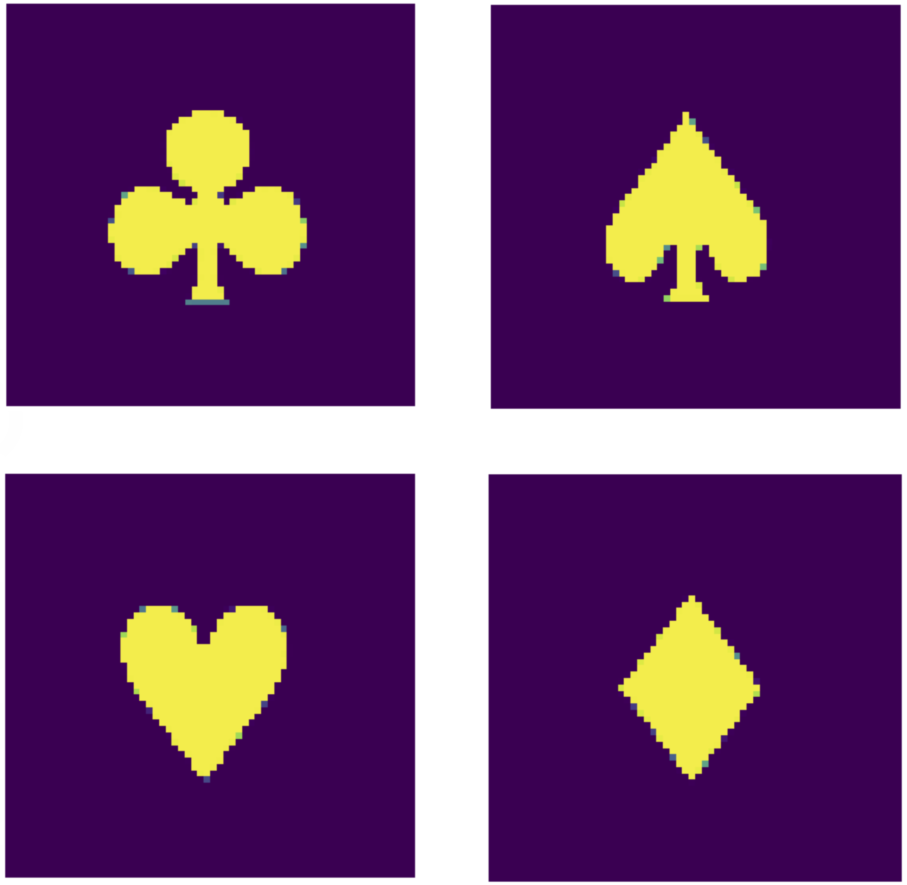
What determines success?



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

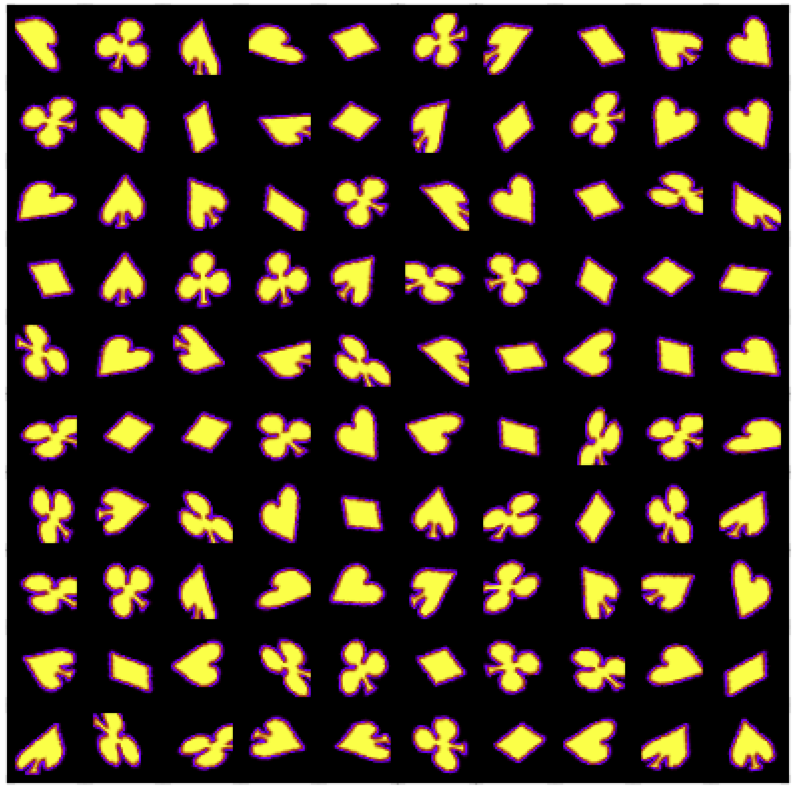
Card data set



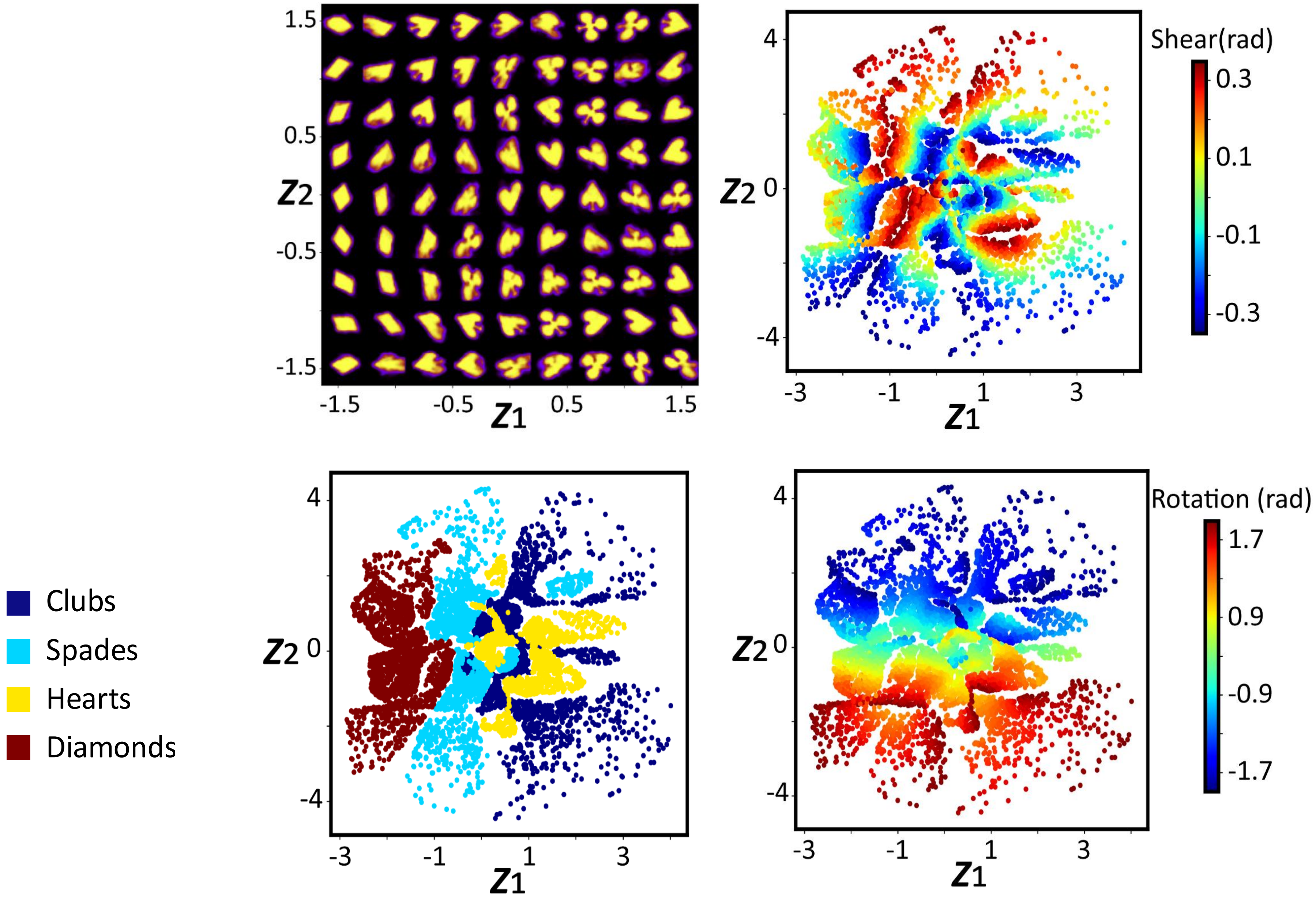
Rotations:
[-120˚, 120˚]

Shear:
[-20˚, 20˚]

Randomly picked samples

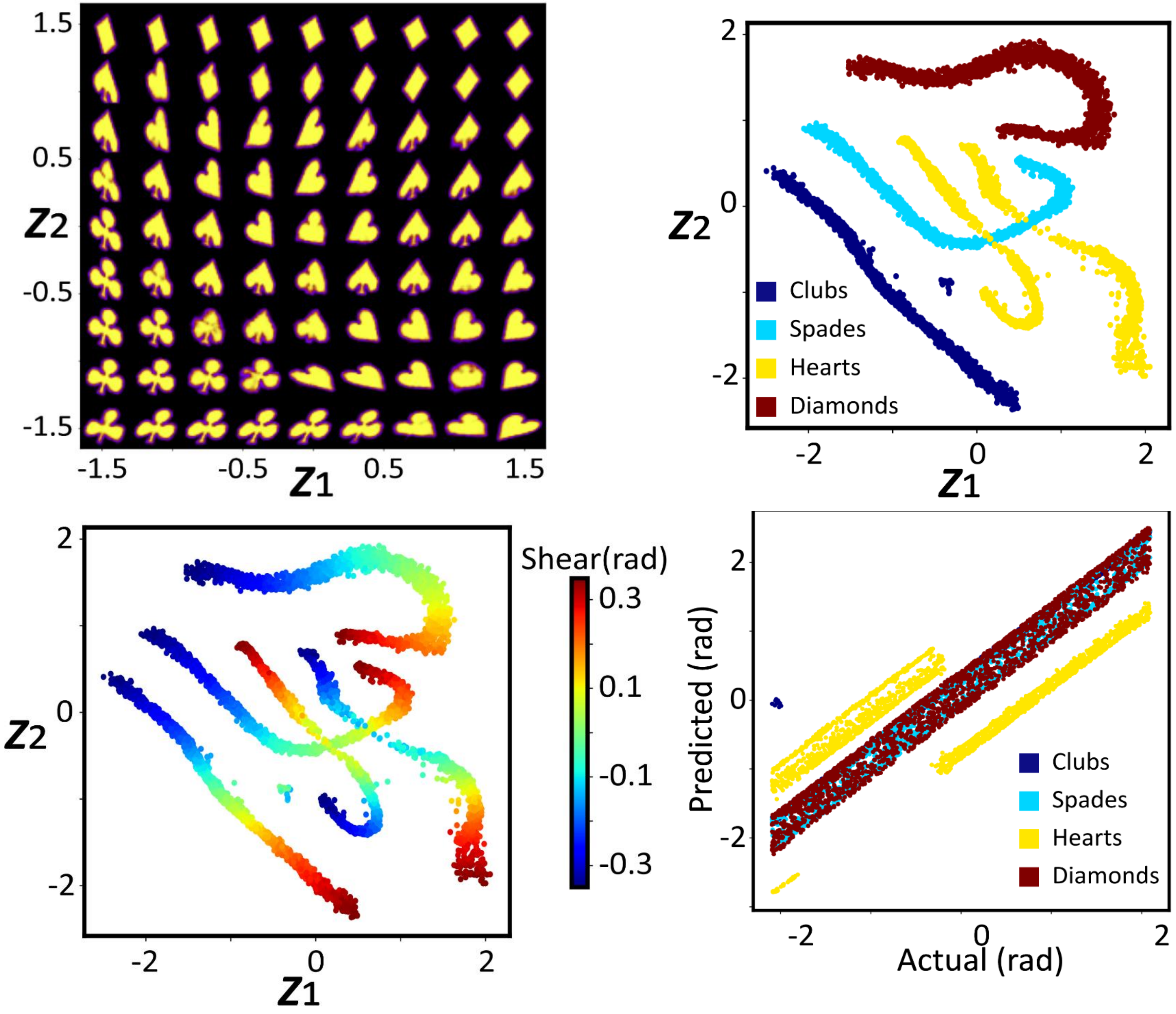


VAE on Cards



1.5
0.5
-0.5
-1.5
z_2
-1.5
-0.5
z_1
0.5
1.5
Shear(rad)
0.3
0.1
-0.1
-0.3
4
0
-4
-3
-1
1
3
Clubs
Spades
Hearts
Diamonds
Rotation (rad)
1.7
0.9
-0.9
-1.7

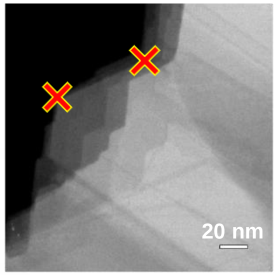
rVAE on Cards

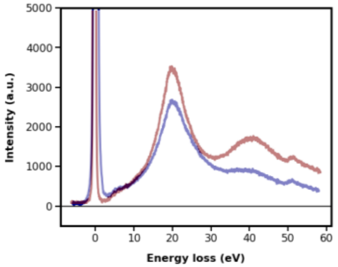


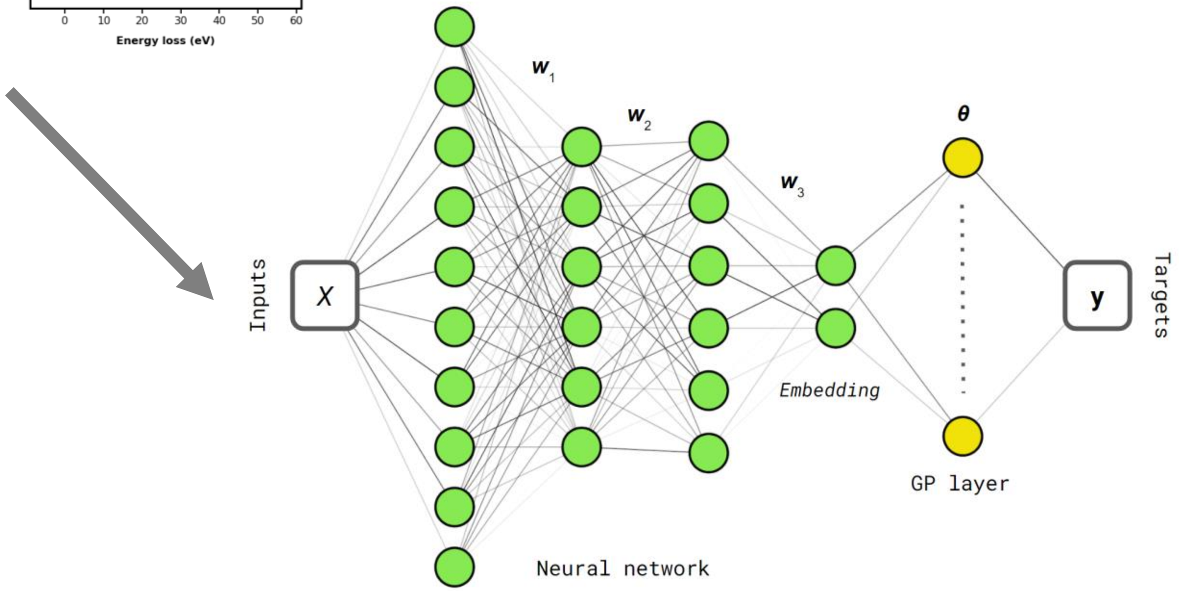
Reminder: Deep Kernel Learning

Specify physics criteria

Active learning







Acquire **structural data**

Measure a **spectrum**

Train DKL model with new data

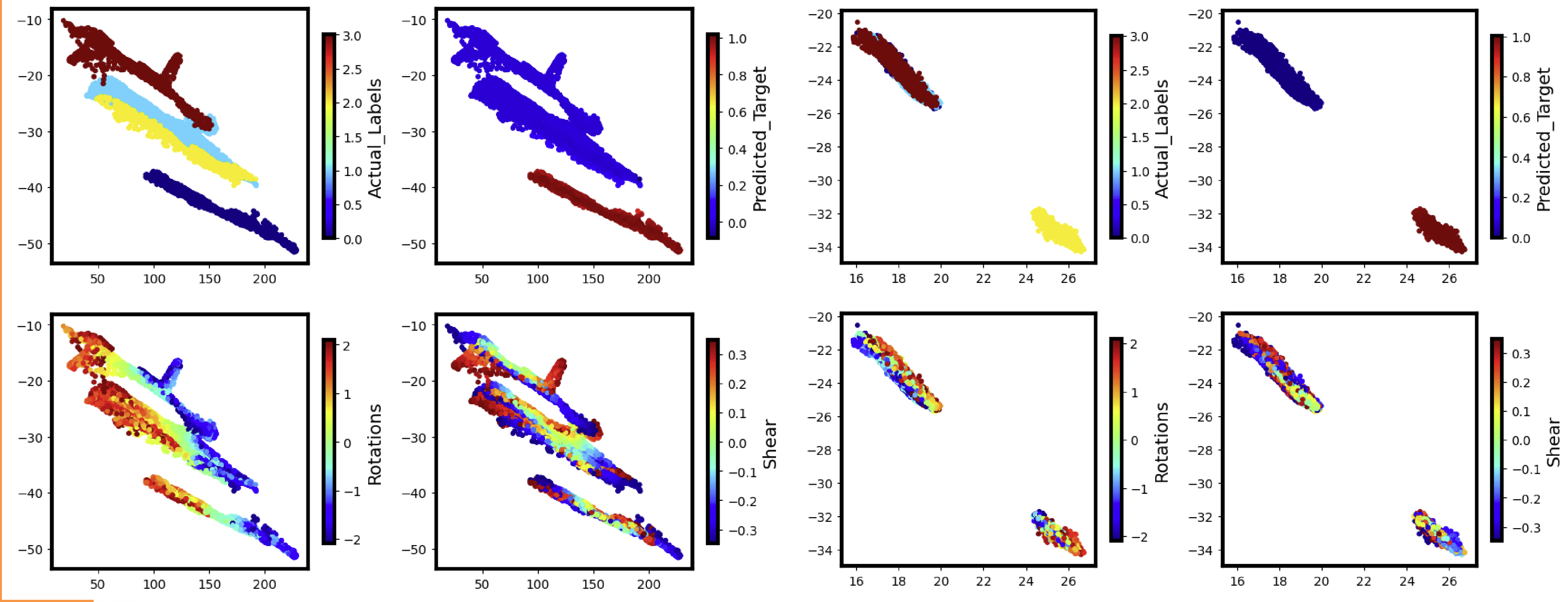
Decide next position (optimize physics criteria)

Allows navigation of the system to search for physics

DKL to predict labels

Clubs

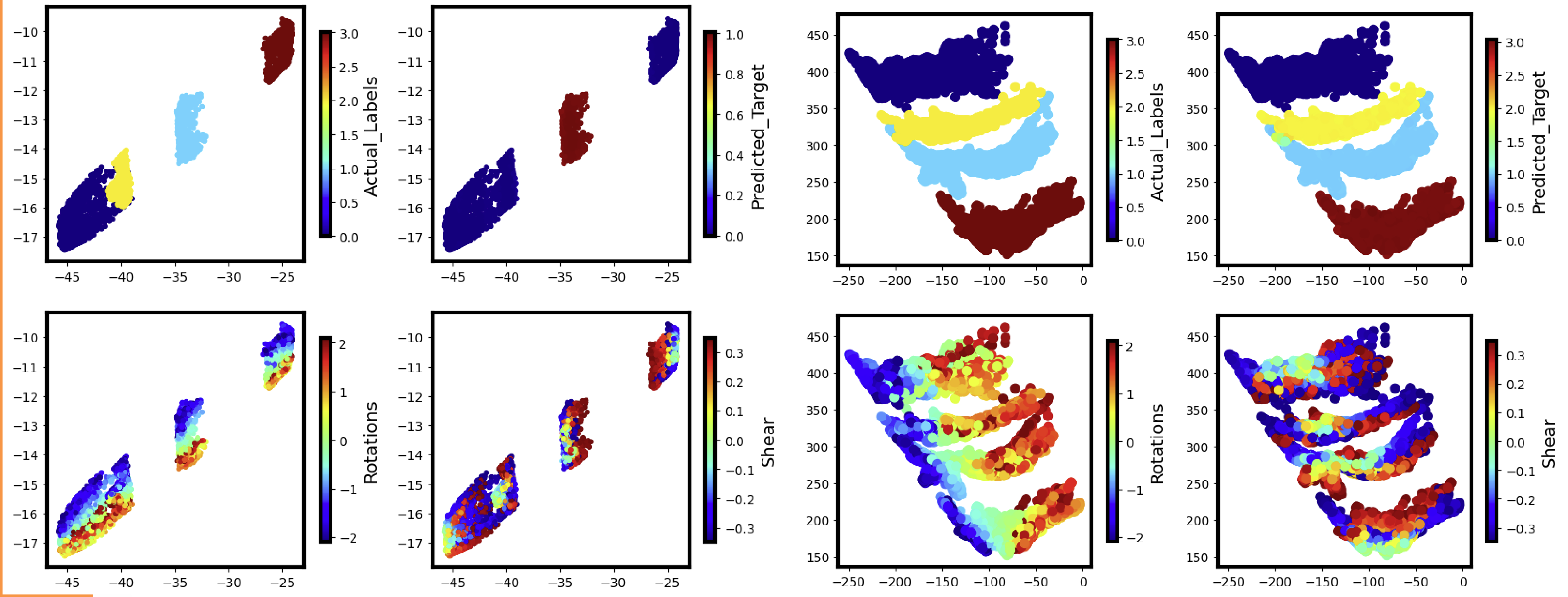
Hearts



DKL to predict labels

Spades

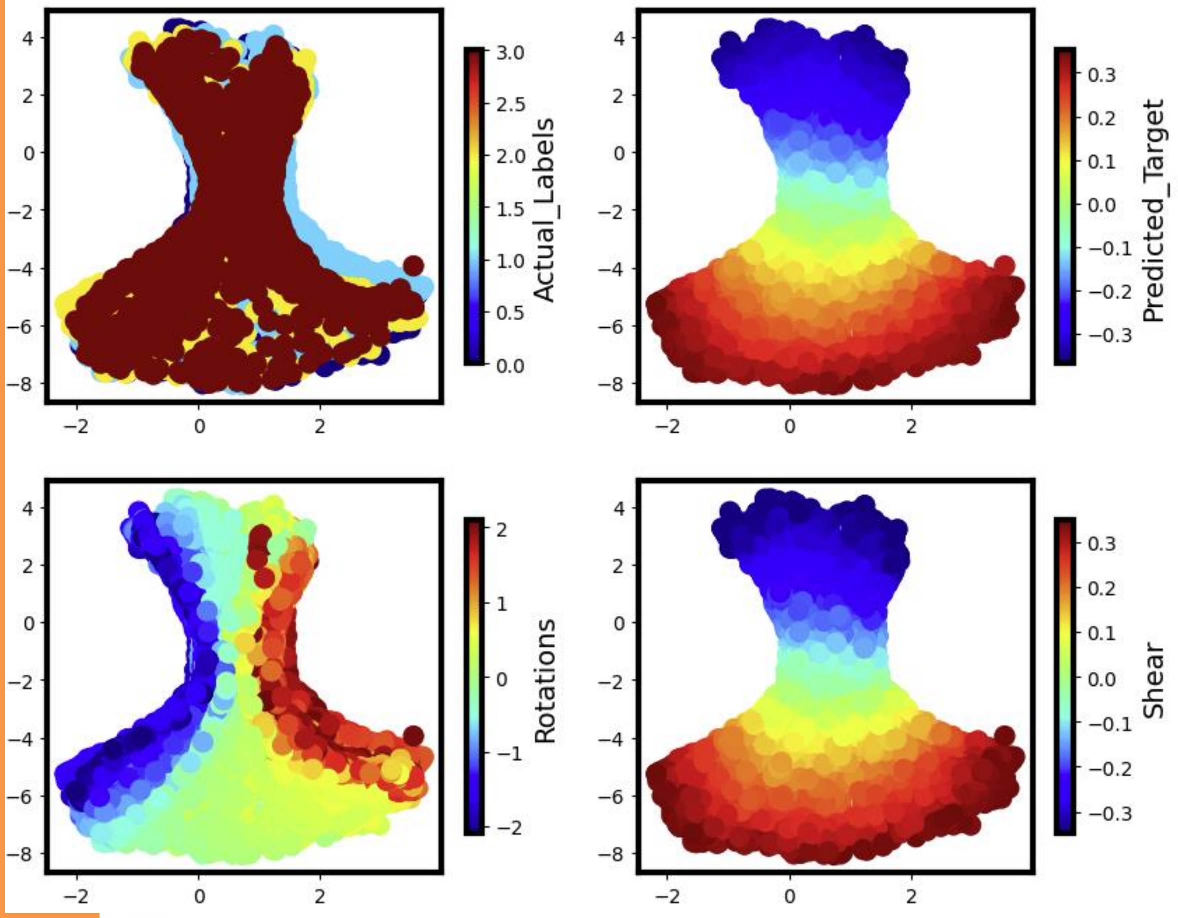
All suits



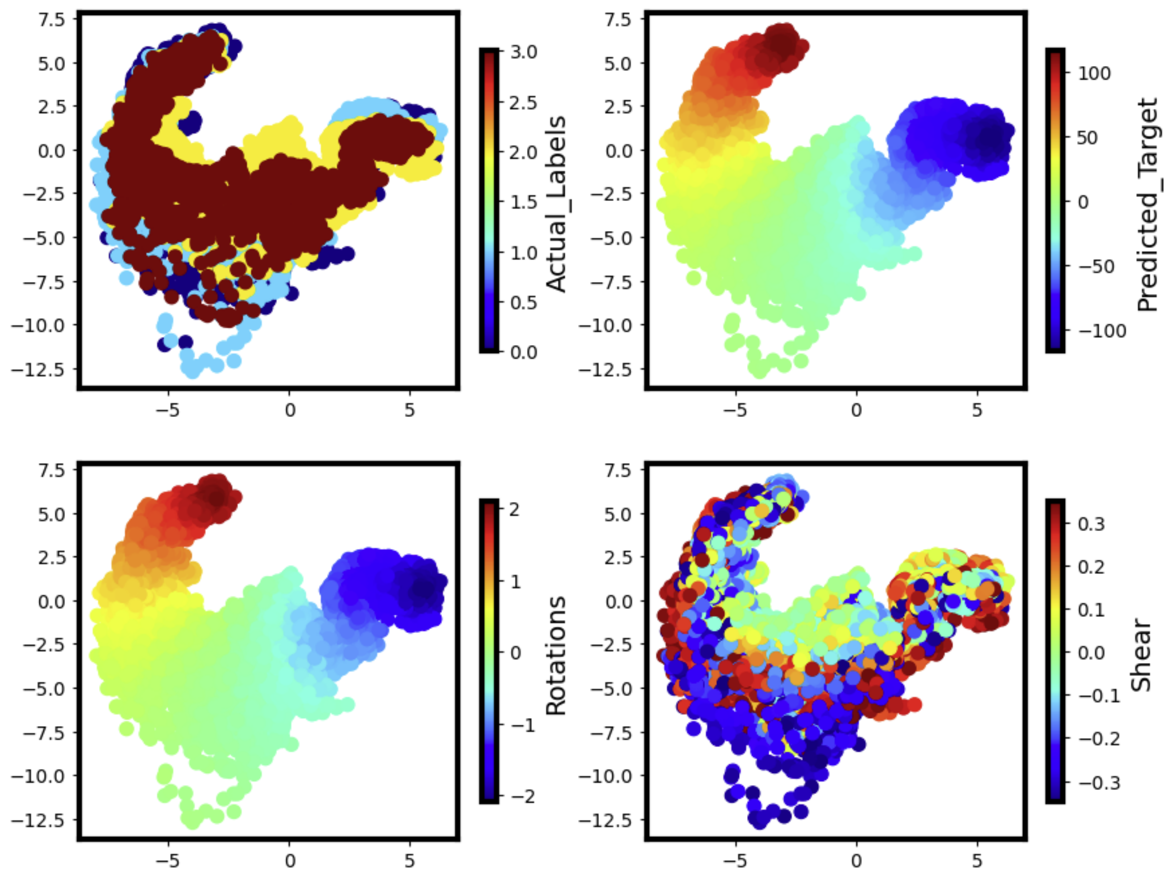
The DKL clearly forms the manifold based on the label!

DKL to predict continuous target function

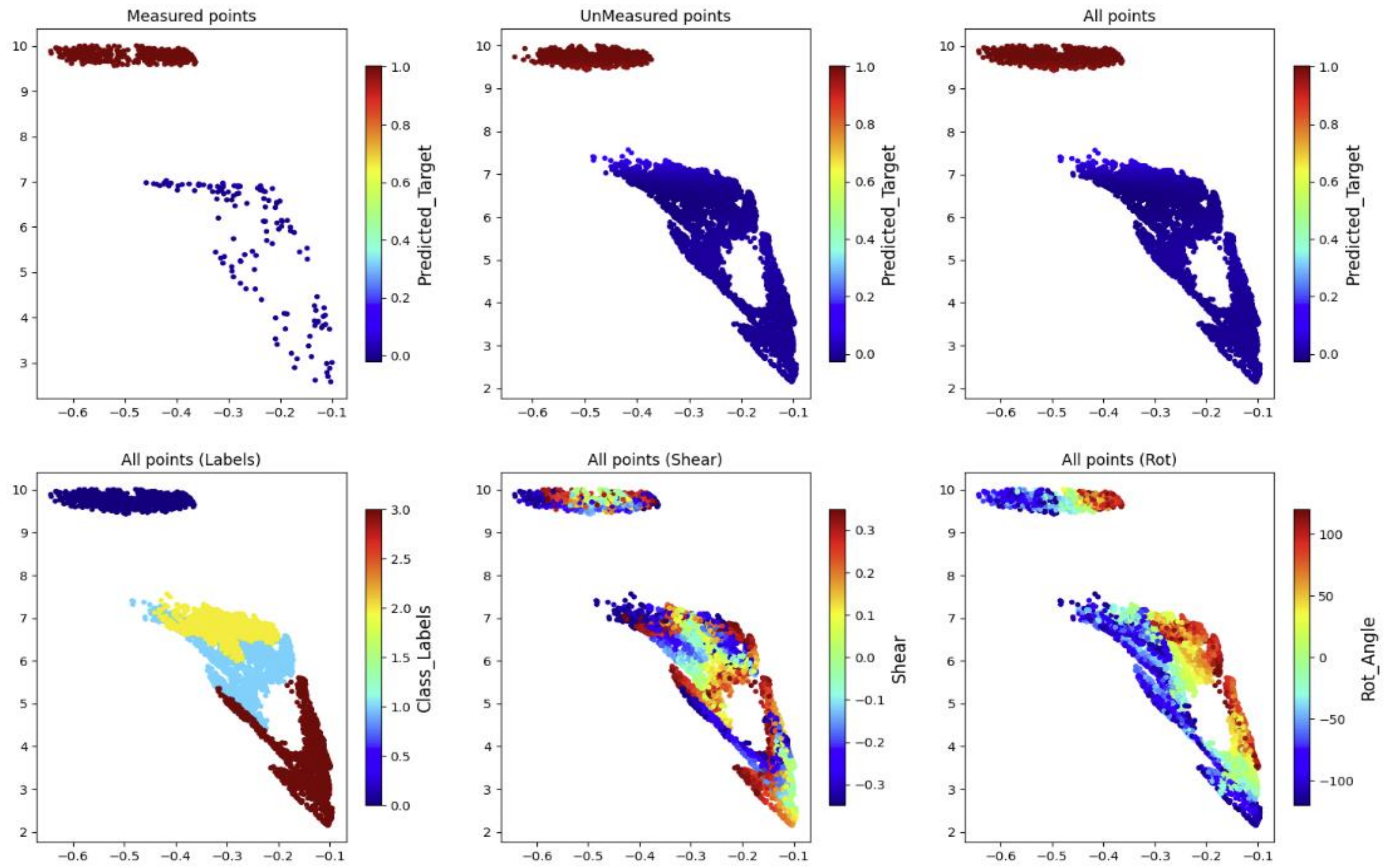
Shear



Rotations

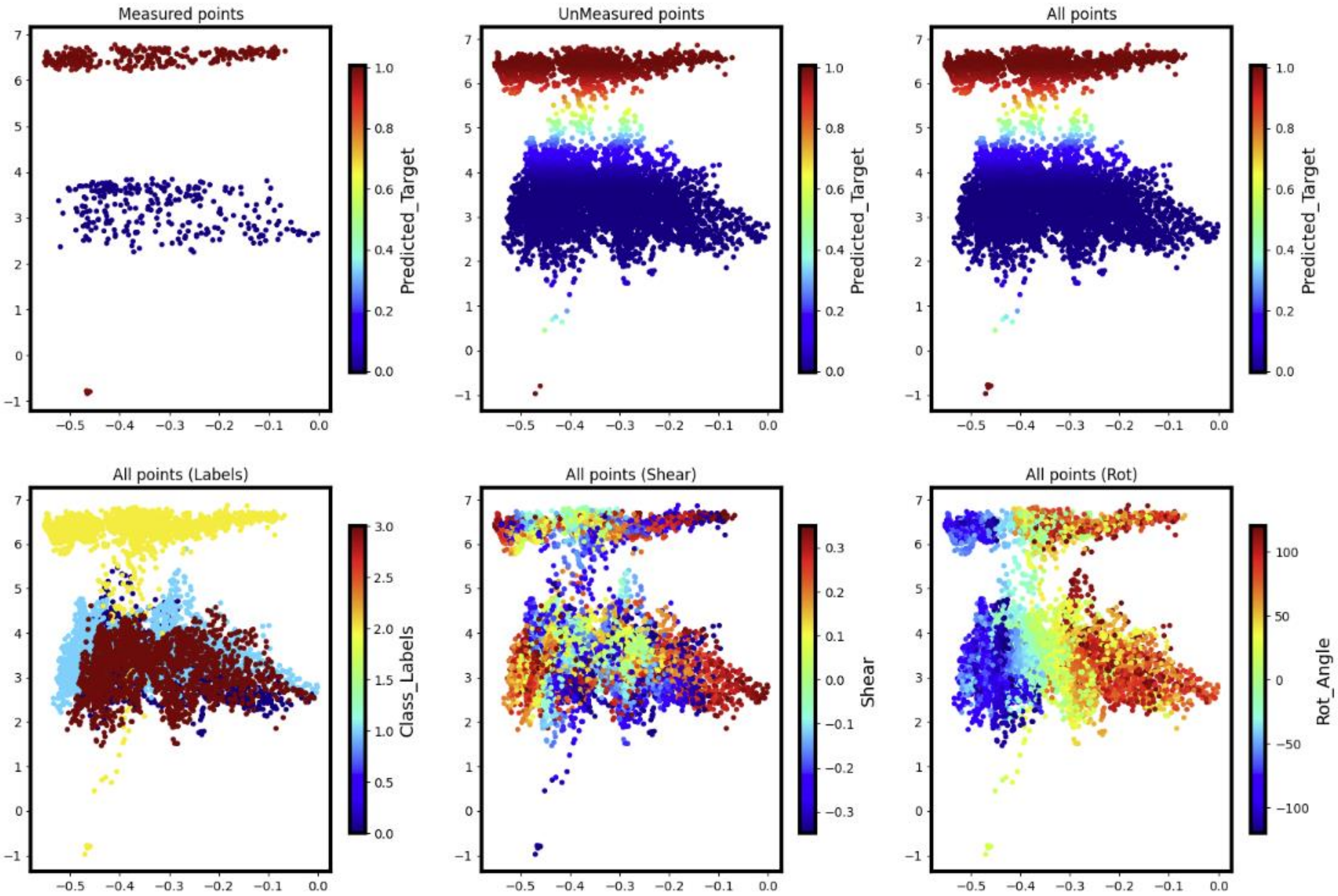


DKL BO: Active Learning

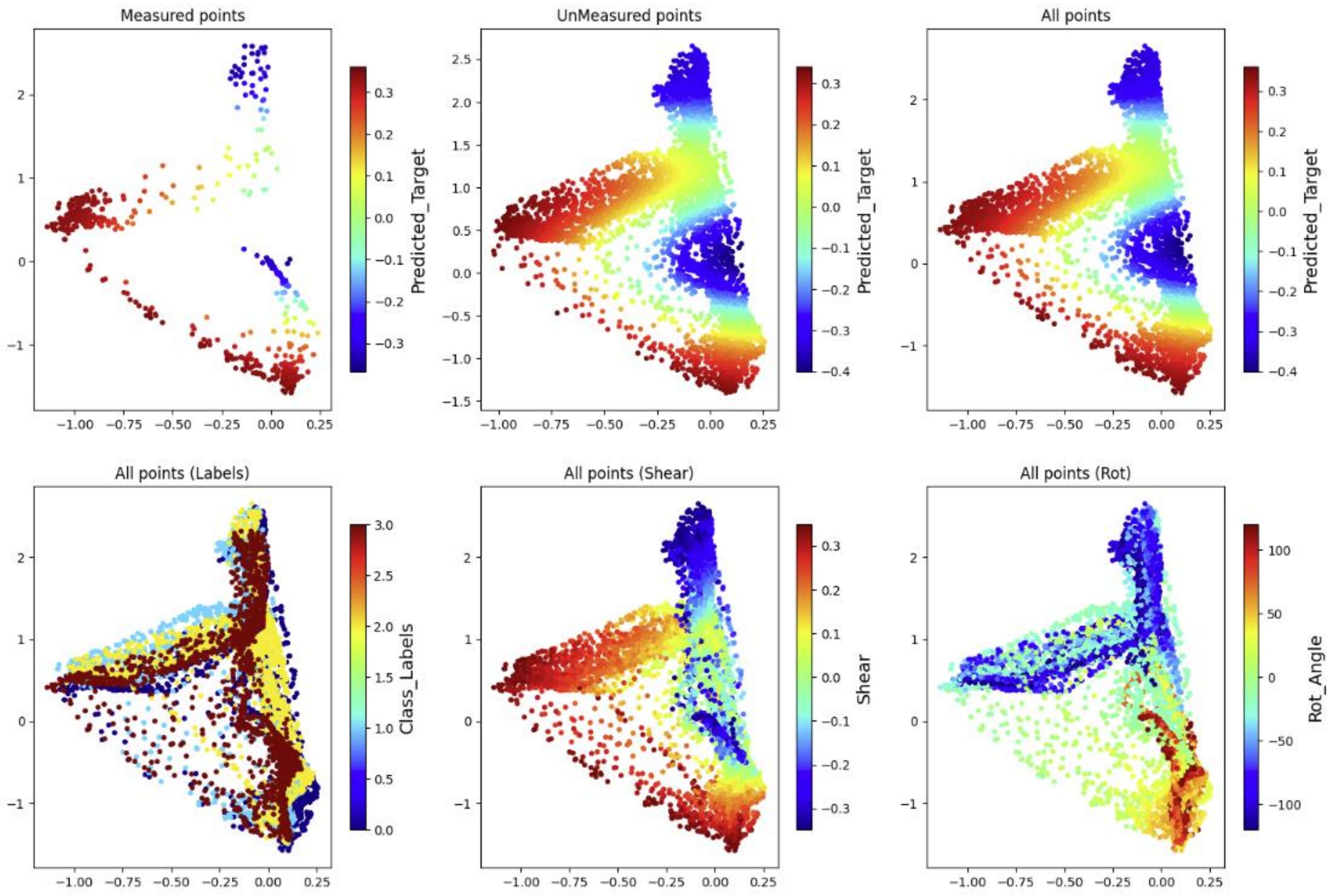


- 100 initialization points and then BO explored 500 points subsequently
- Acquisition function: $\mu + 10\sigma$

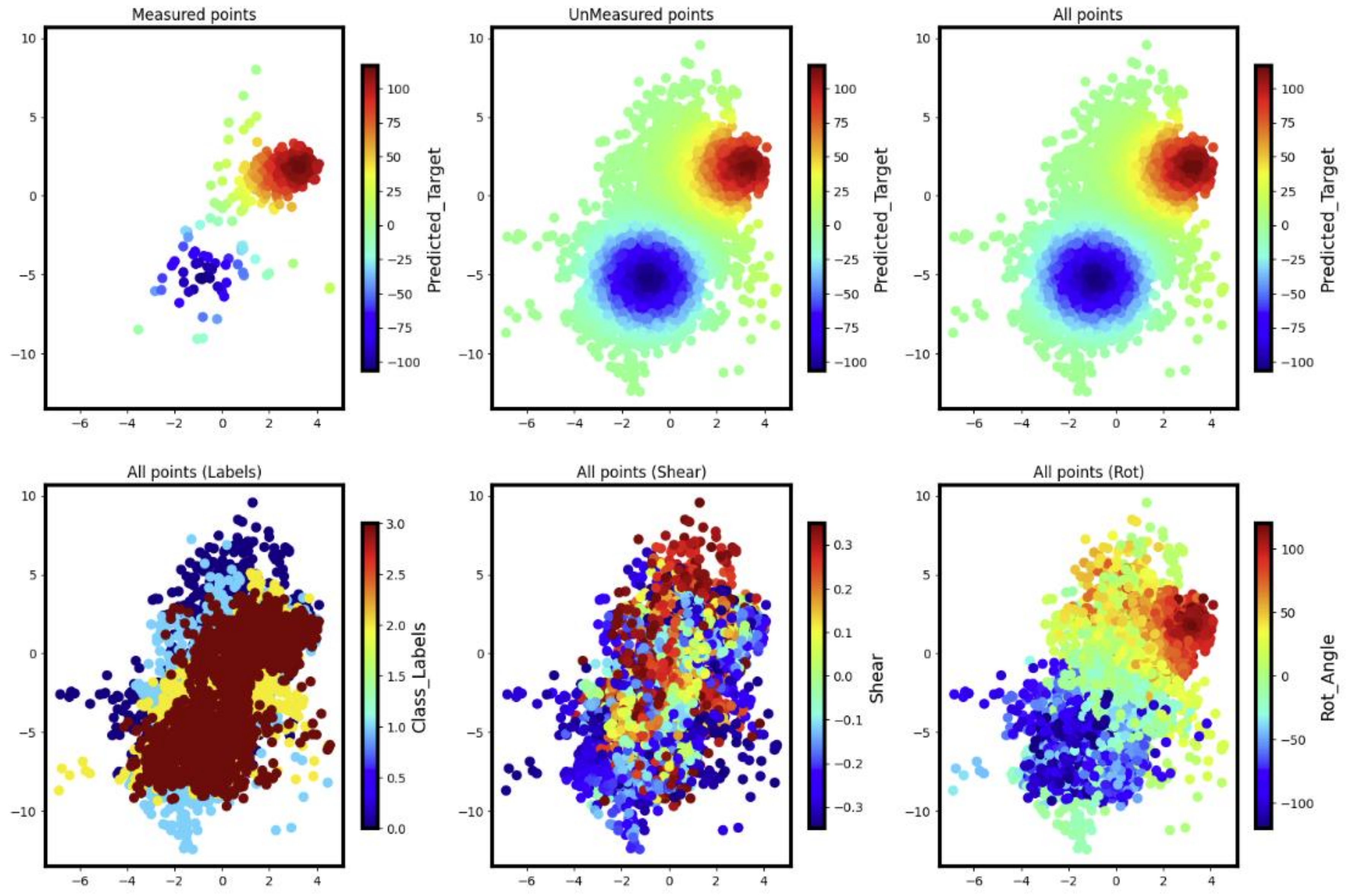
DKL BO: Hearts



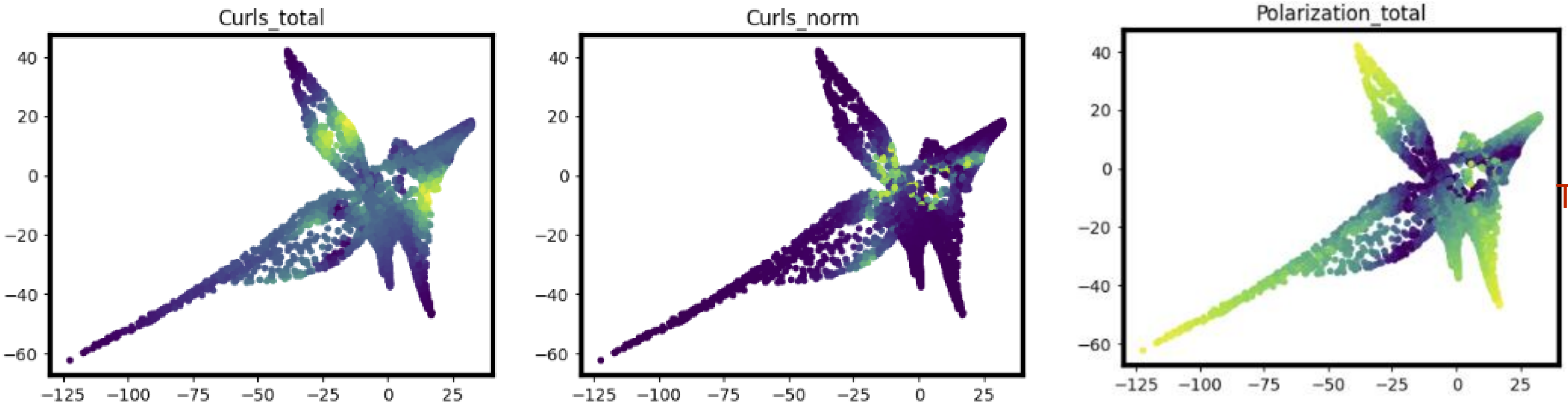
DKL BO: Shear


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

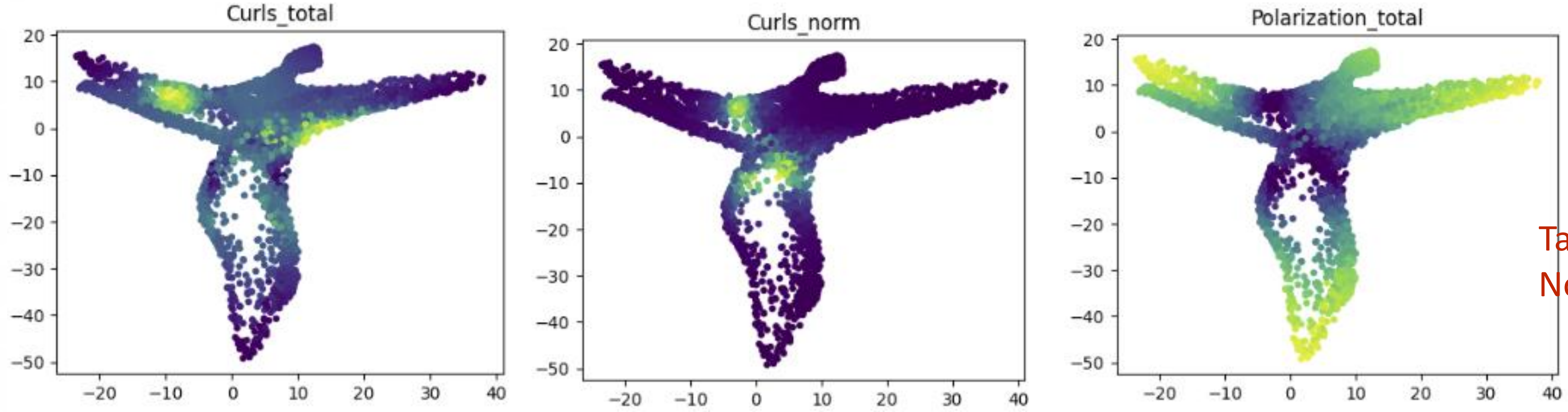
DKL BO: Rotations



DKL on FerroSIM: Static

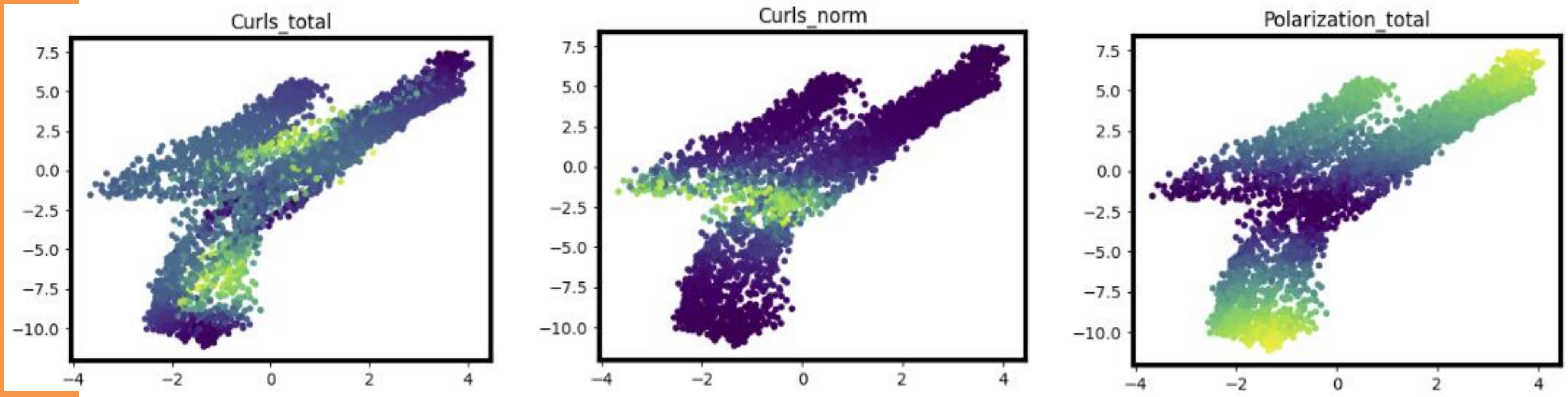


Target function: Curl



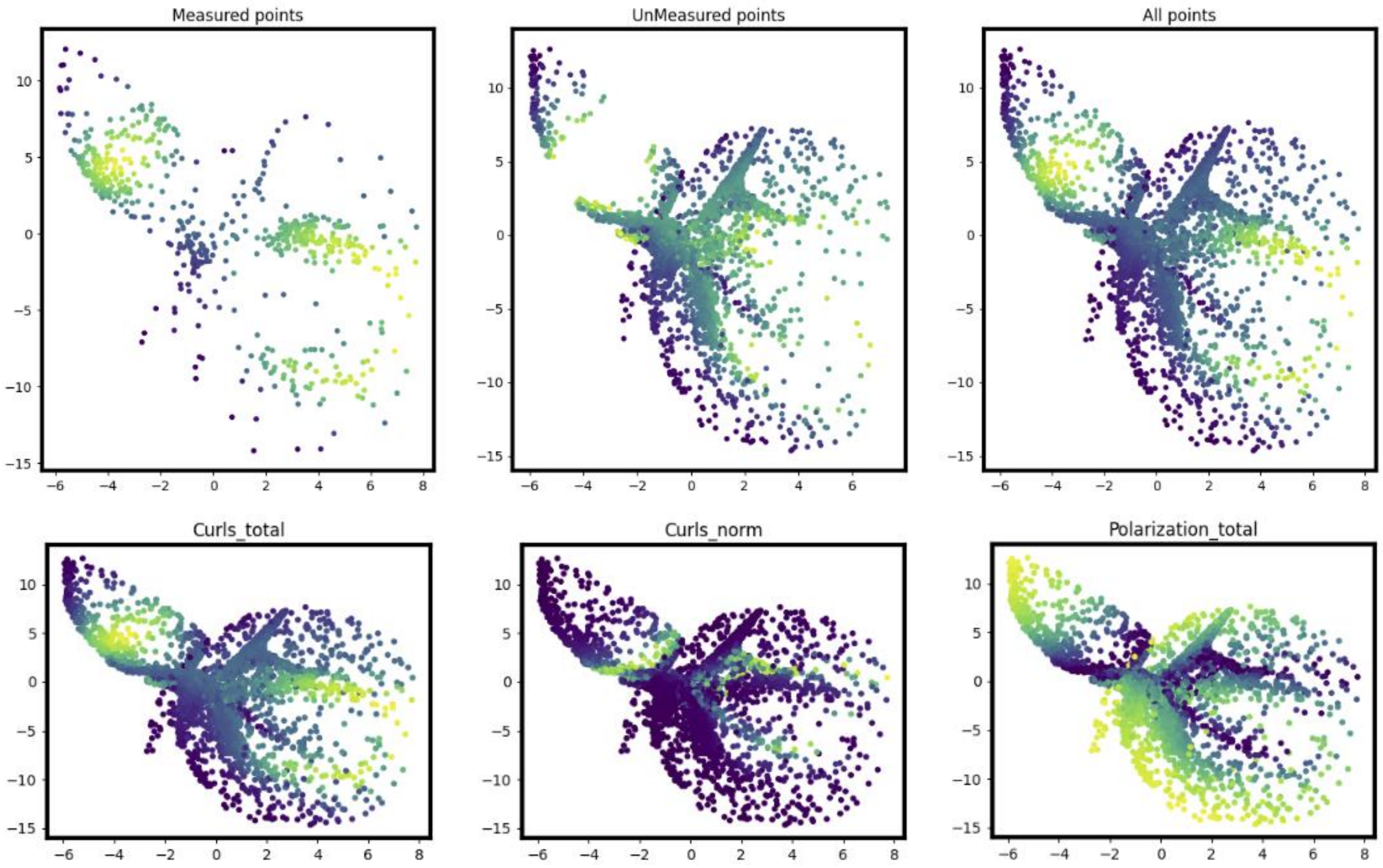
Target function: Normalized curl

DKL on FerroSIM: Active Learning



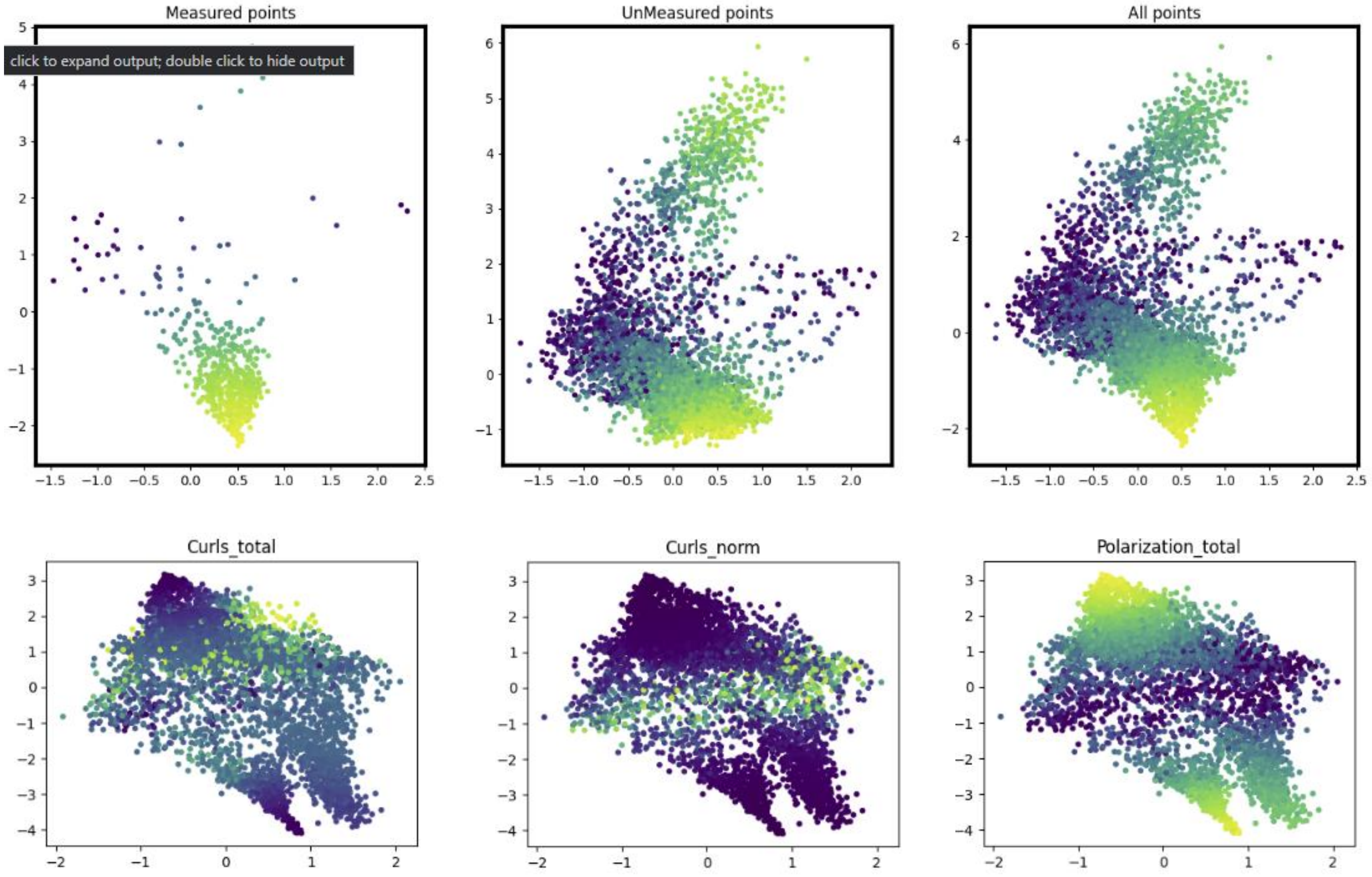
Target function: Polarization

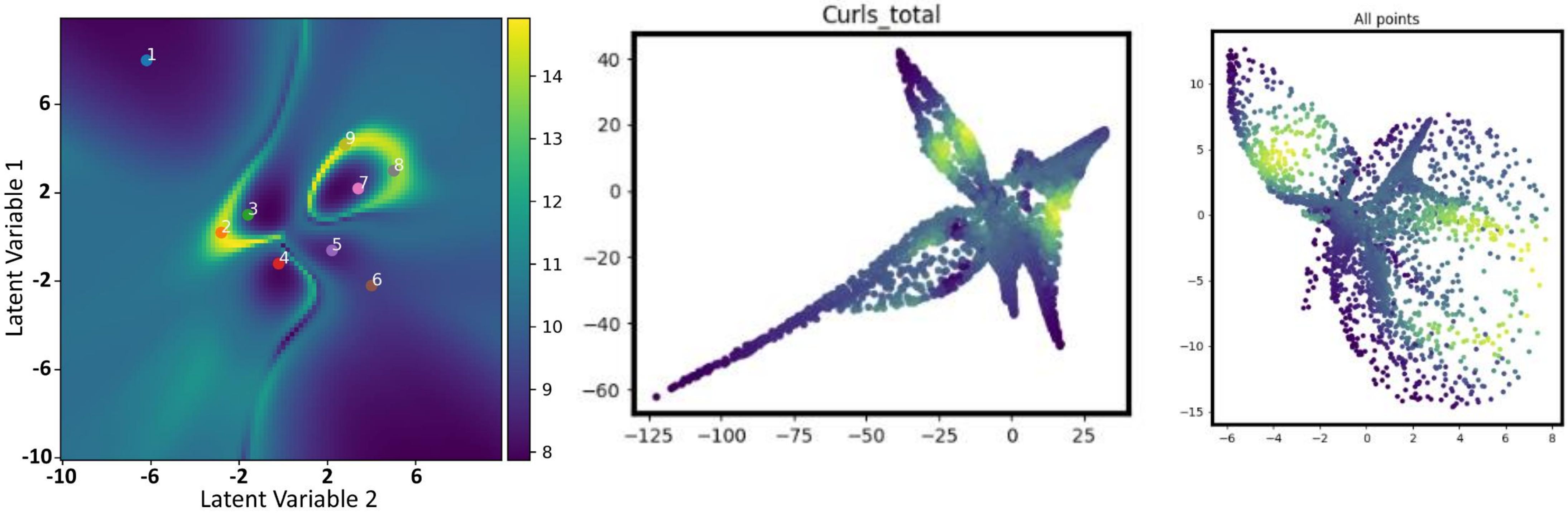
DKL BO: Curl


Measured points
UnMeasured points
All points
Curls_total
Curls_norm
Polarization_total

DKL BO: Polarization

the latent space





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!