

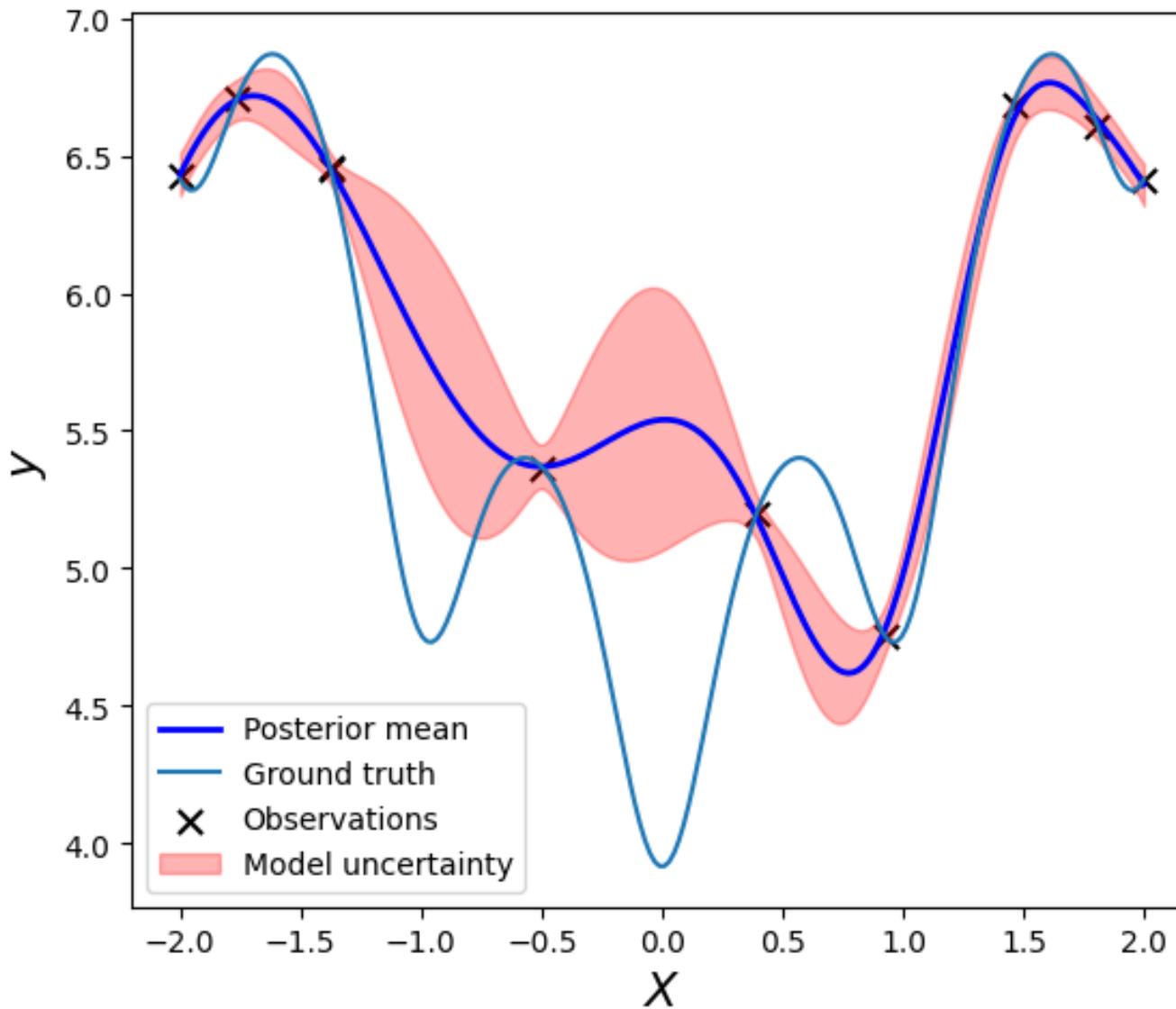
# Deep Kernel Learning - I

Sergei V. Kalinin

# What have we learned from lectures on GP/BO

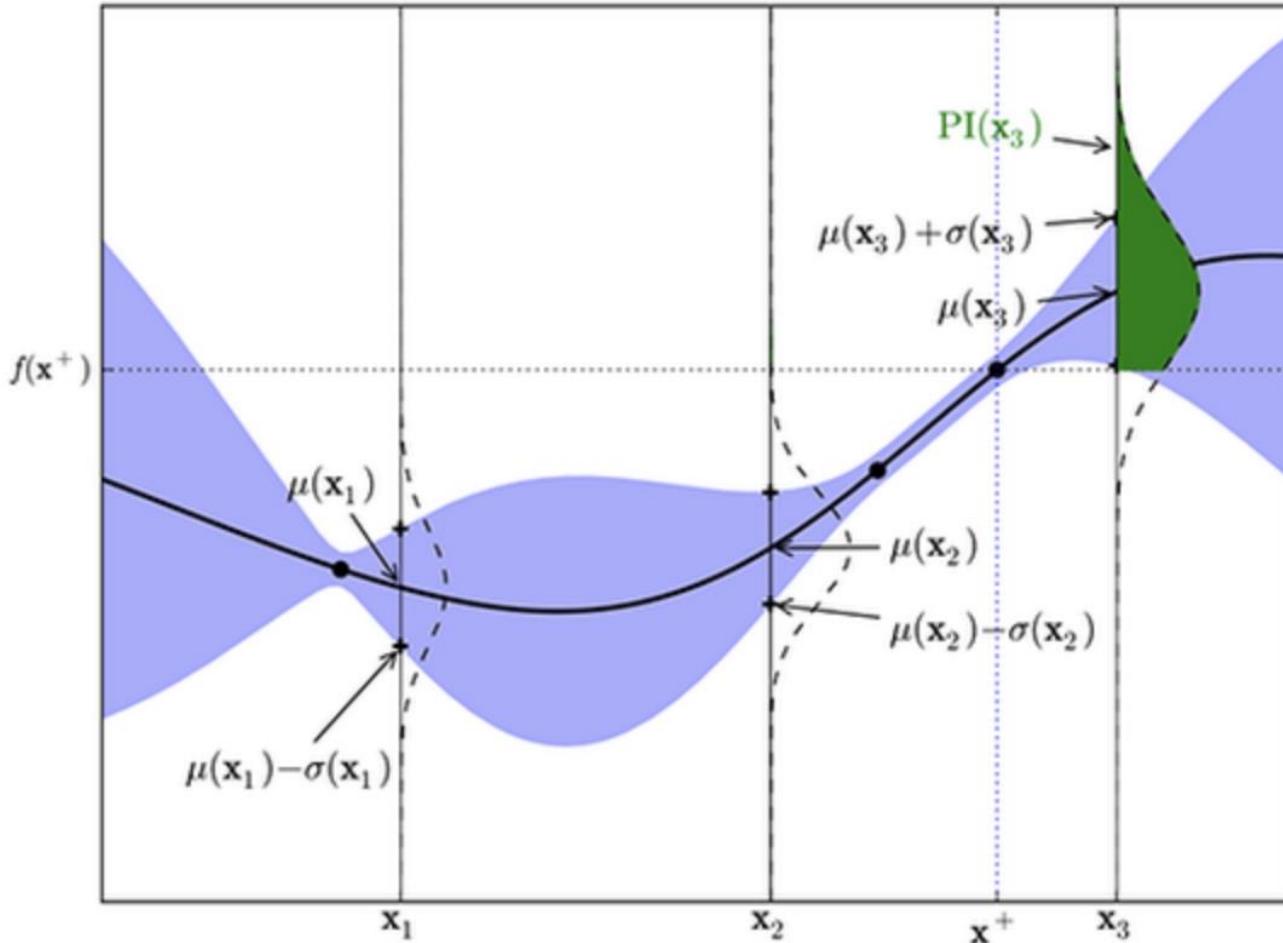
- Gaussian Process
- Kernel and kernel parameters
- Kernel Priors
- Noise Priors
- Mean function and priors
- Posteriors
- Bayesian Inference
- Bayesian optimization
- Acquisition function

# Gaussian Process and Bayesian Inference



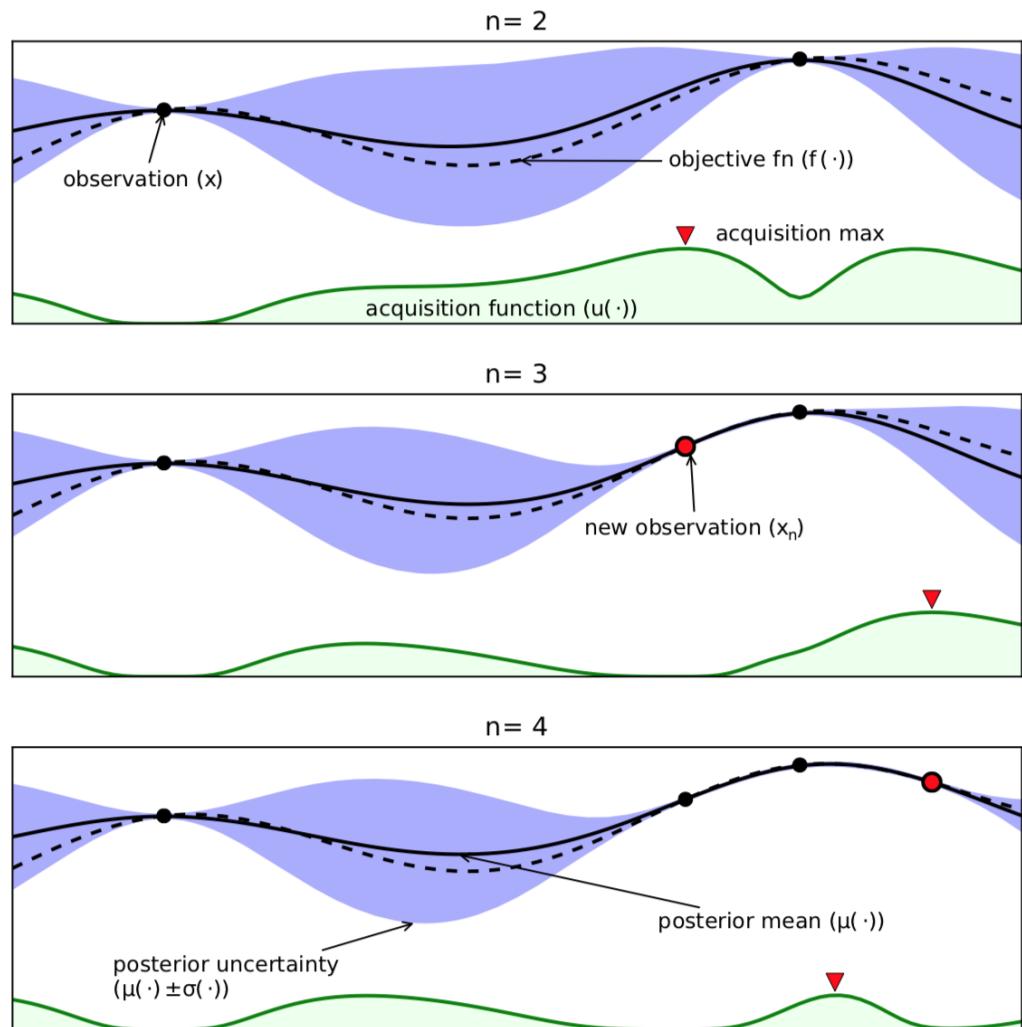
- We have some measurements in space  $X$ , and we want to maximize some property  $f(X)$ .
- We create surrogate model: function and uncertainty over full parameter space based on measurements
- Gaussian Process: purely data driven
- Bayesian Inference: known model and some idea on parameters
- Structured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

# Acquisition Functions (Policies)

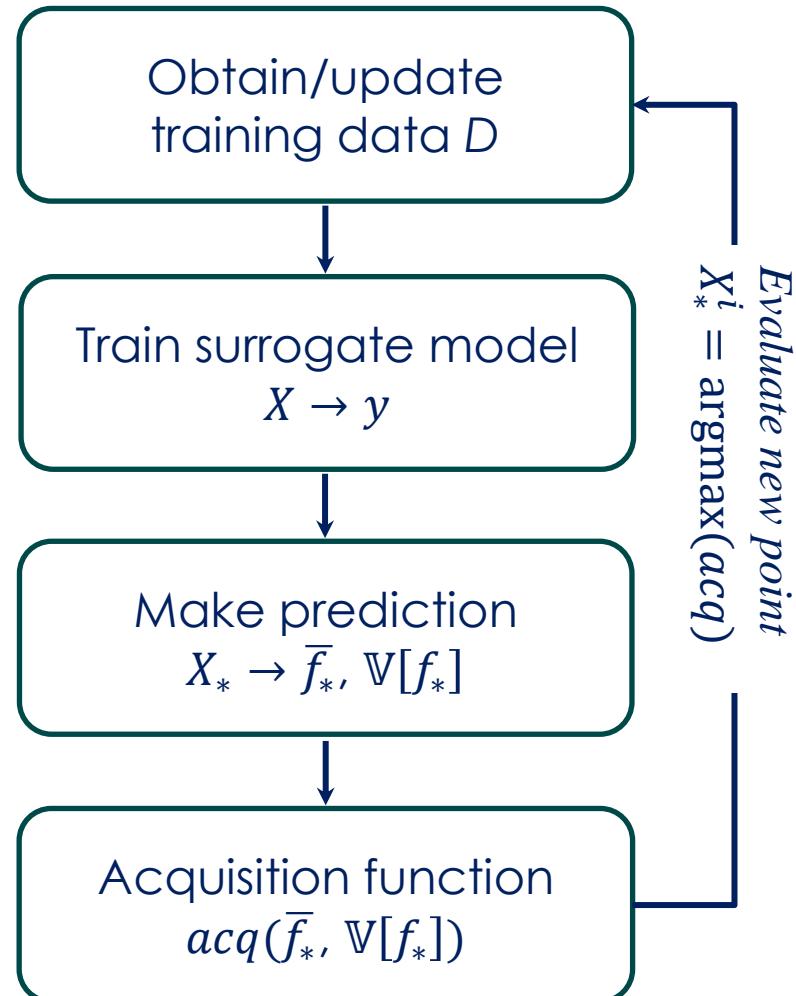


- 1. Upper confidence bound:** simplest possible - just take the upper confidence bound from the prediction
- 2. Probability of Improvement:** Integral from current functional maximum to upper limit of distribution as test point
- 3. Expected Improvement:** Instead of probability of improvement, we want to maximize the expected increase in the function value
- 4. There are (always) more...**

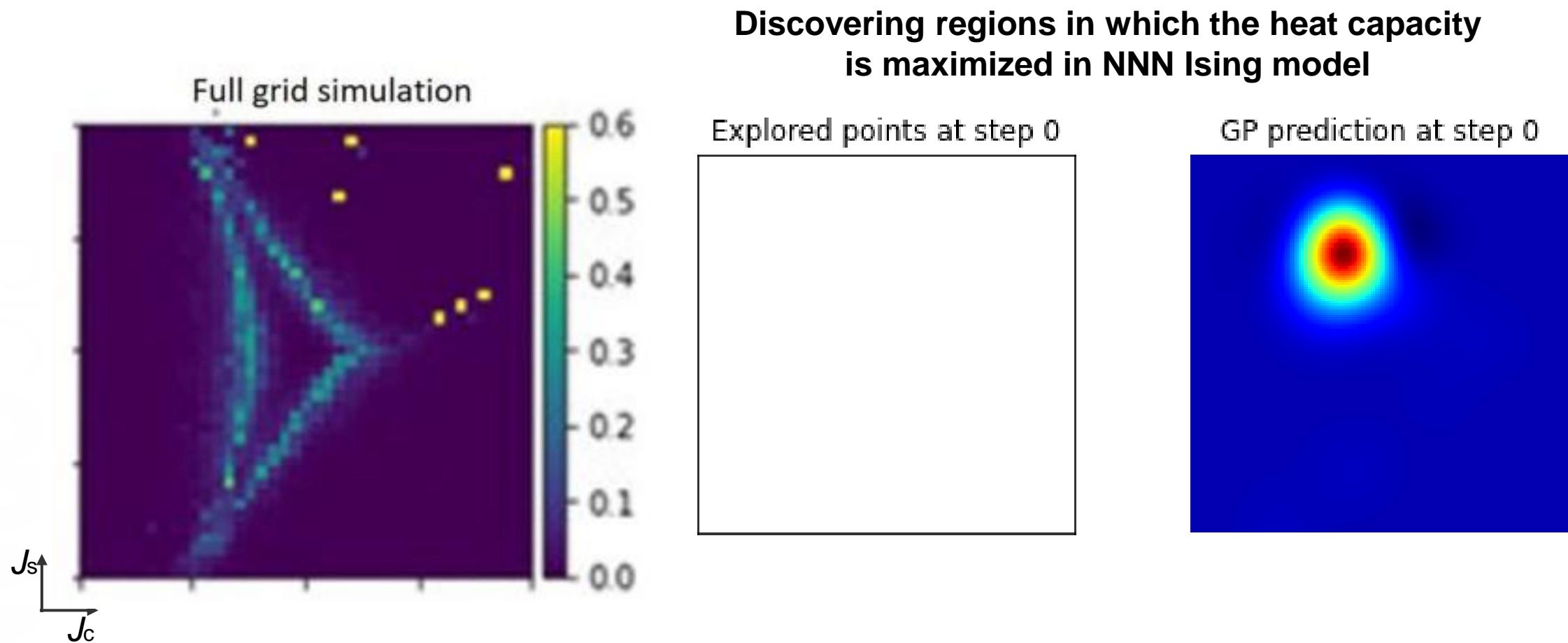
# Bayesian Optimization



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



# Bayesian Optimization for Physical Discovery



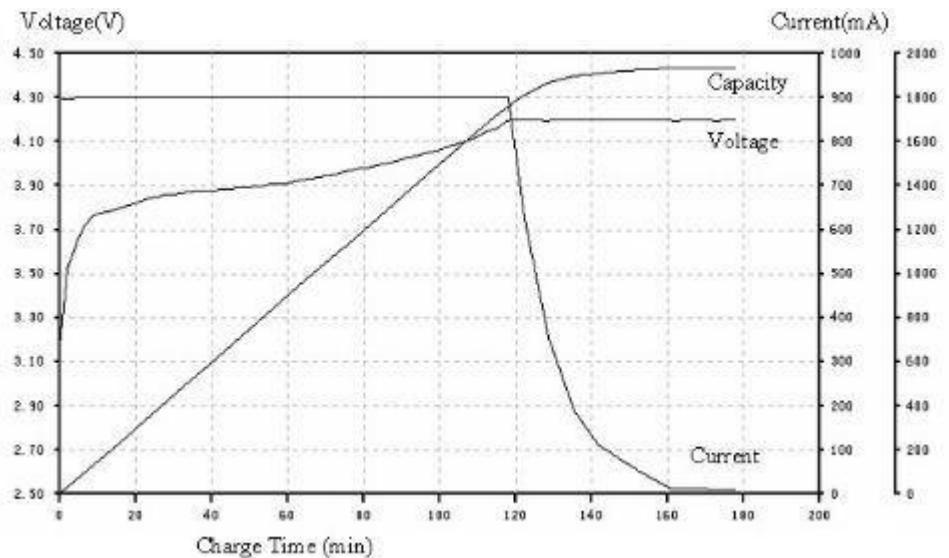
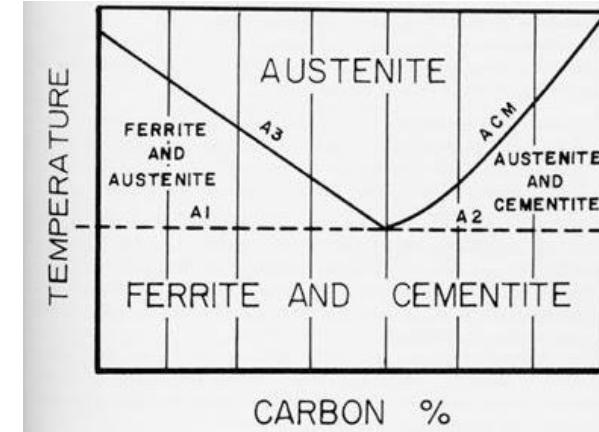
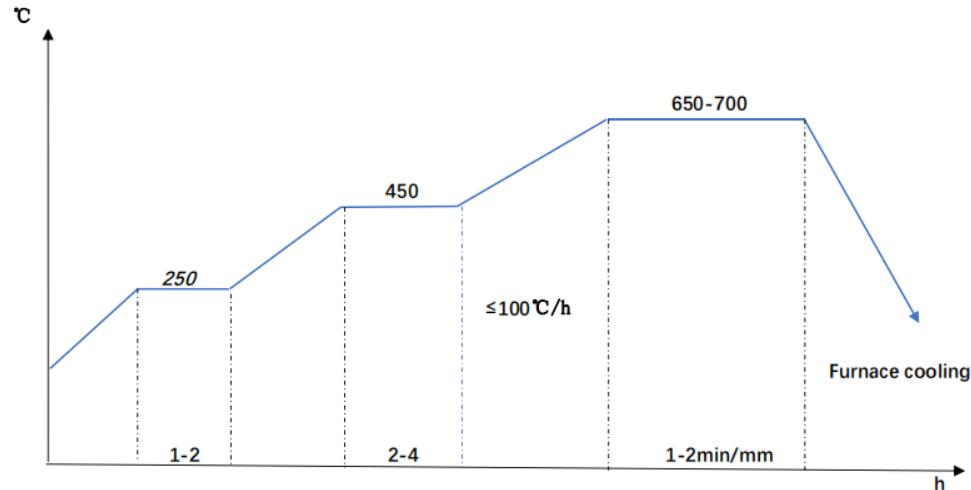
# What is the limitation of the GP/BO?

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

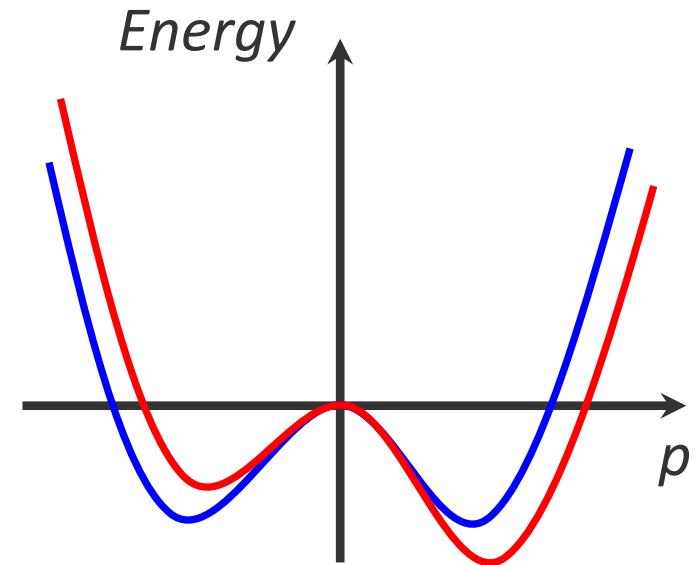
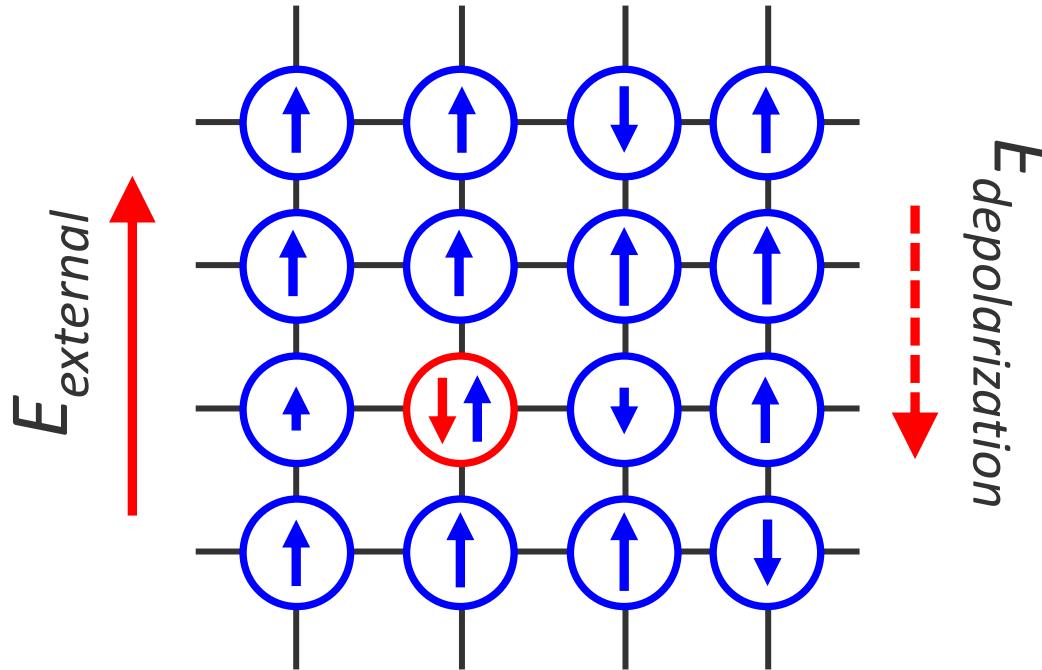
# Making materials: process trajectories



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

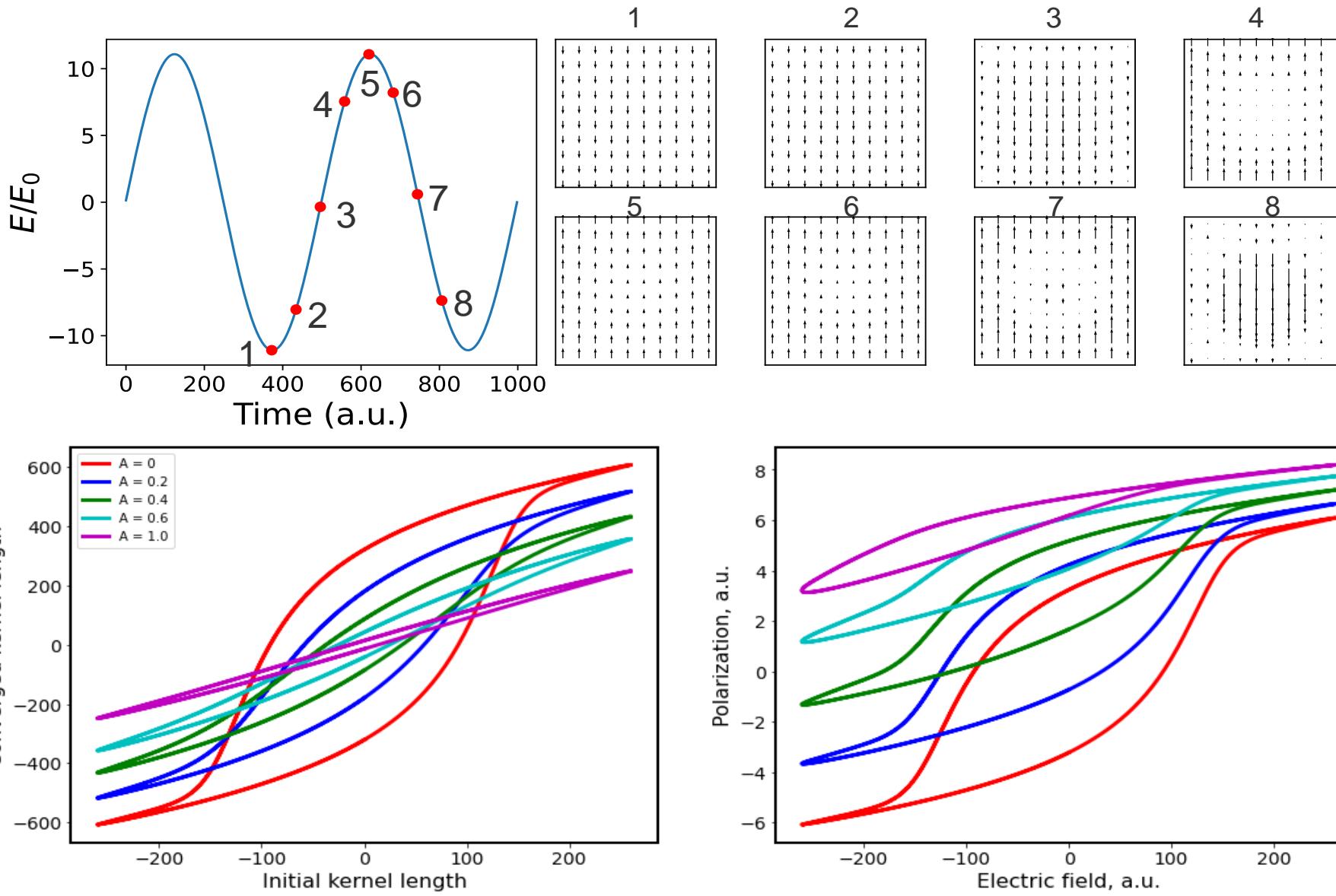
How do we optimize trajectories if we have (a) only limited or no mechanistic information, (b) our experimental budgets are limited, but (c) we have some access to domain expertise?

# 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:
  - $F_{ij} = \alpha_1 (p_{x_{ij}}^2 + p_{y_{ij}}^2) + \alpha_2 (p_{x_{ij}}^4 + p_{y_{ij}}^4) + \alpha_3 p_{x_{ij}}^2 p_{y_{ij}}^2 - E_{loc_{x_{ij}}} p_{x_{ij}} - E_{loc_{y_{ij}}} p_{y_{ij}}$
  - 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  $\frac{d p_{i,j}}{dt} = -\frac{\partial F}{\partial p_{i,j}}$

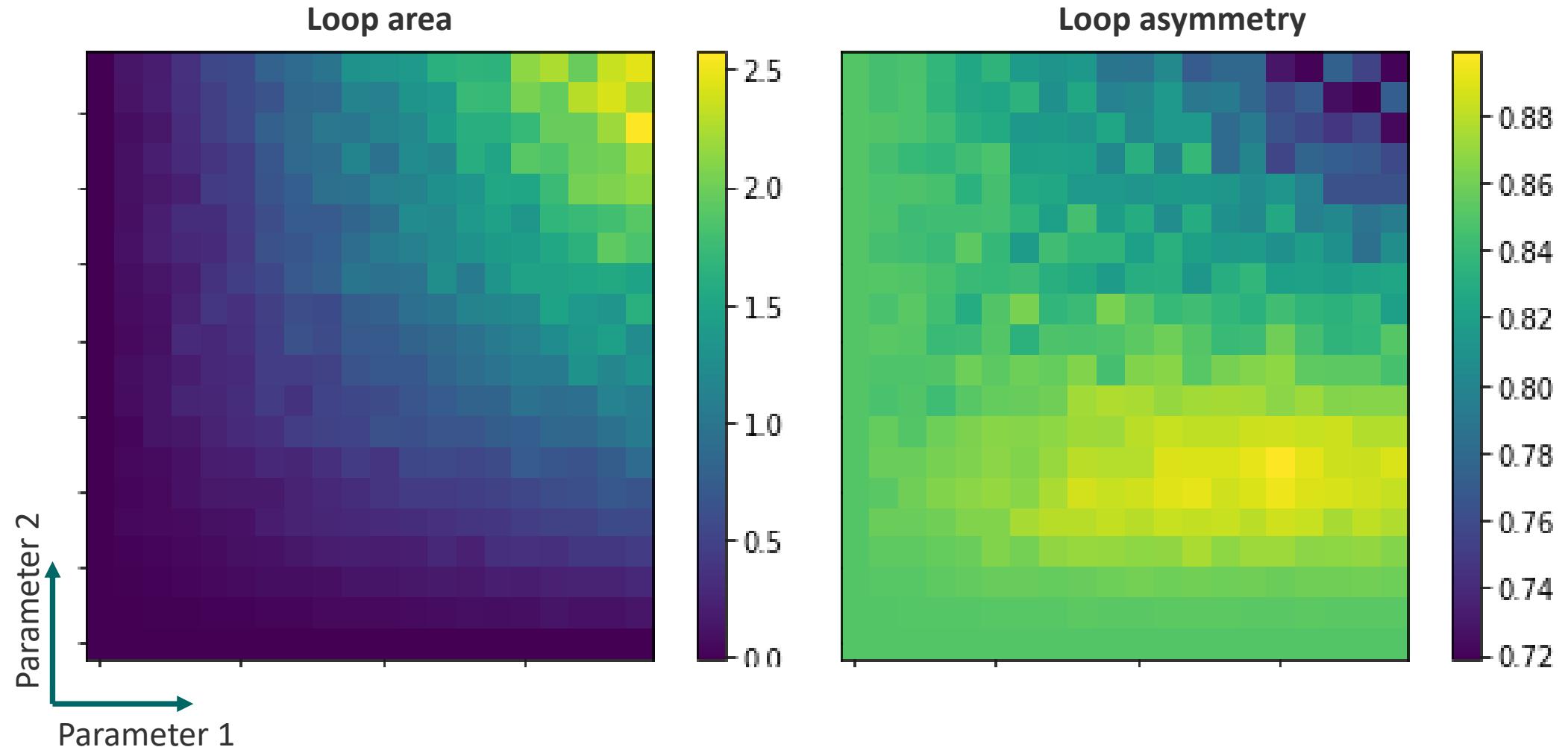
# Microstates and Macroscopic Observables



# Global response vs. model parameters

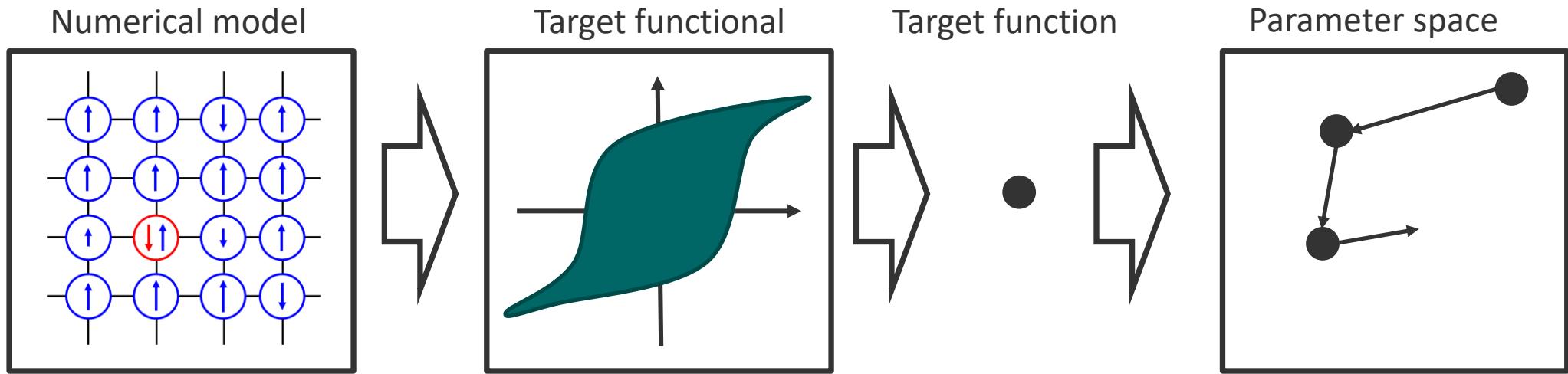
Parameter space 1: Hamiltonian

Parameter space 2: Field history



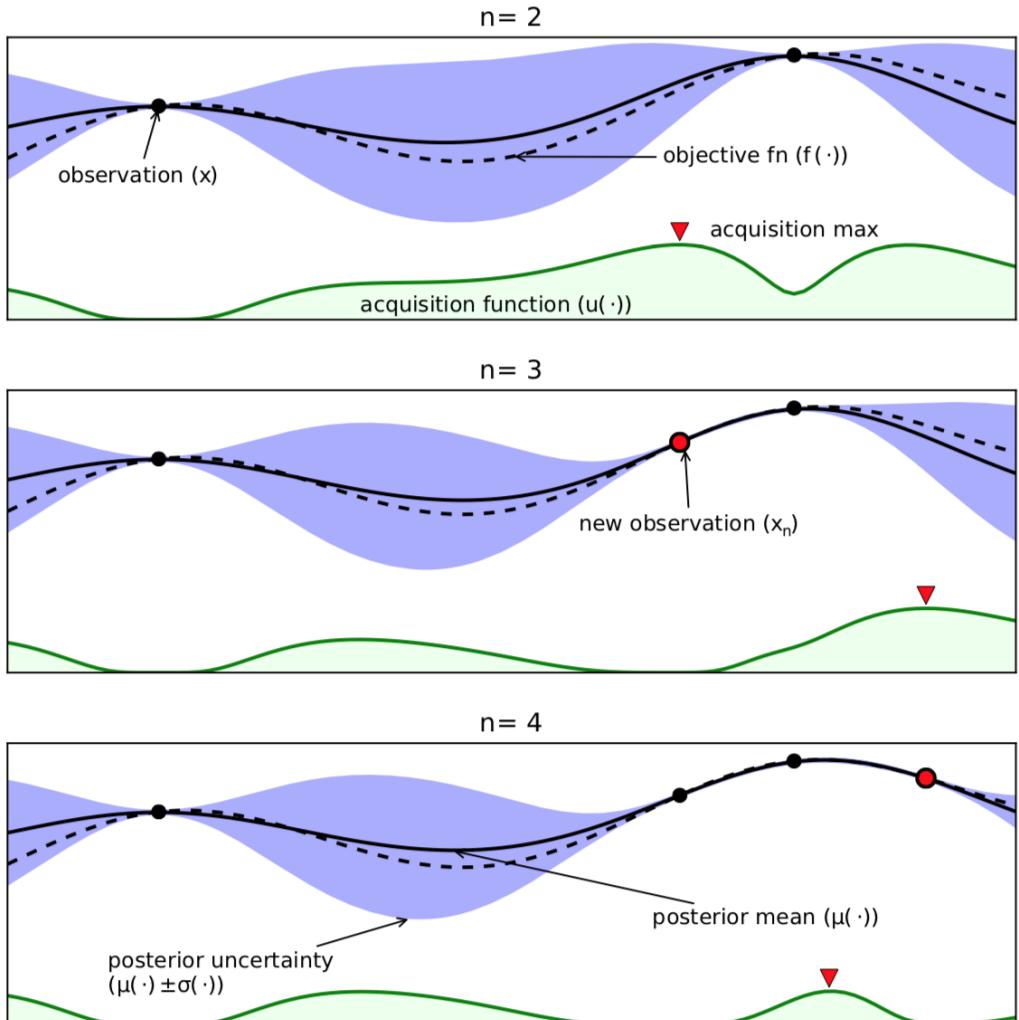
For small dimensional parameter spaces, we can evaluate global responses via the grid search

# Can we do better than grid search?



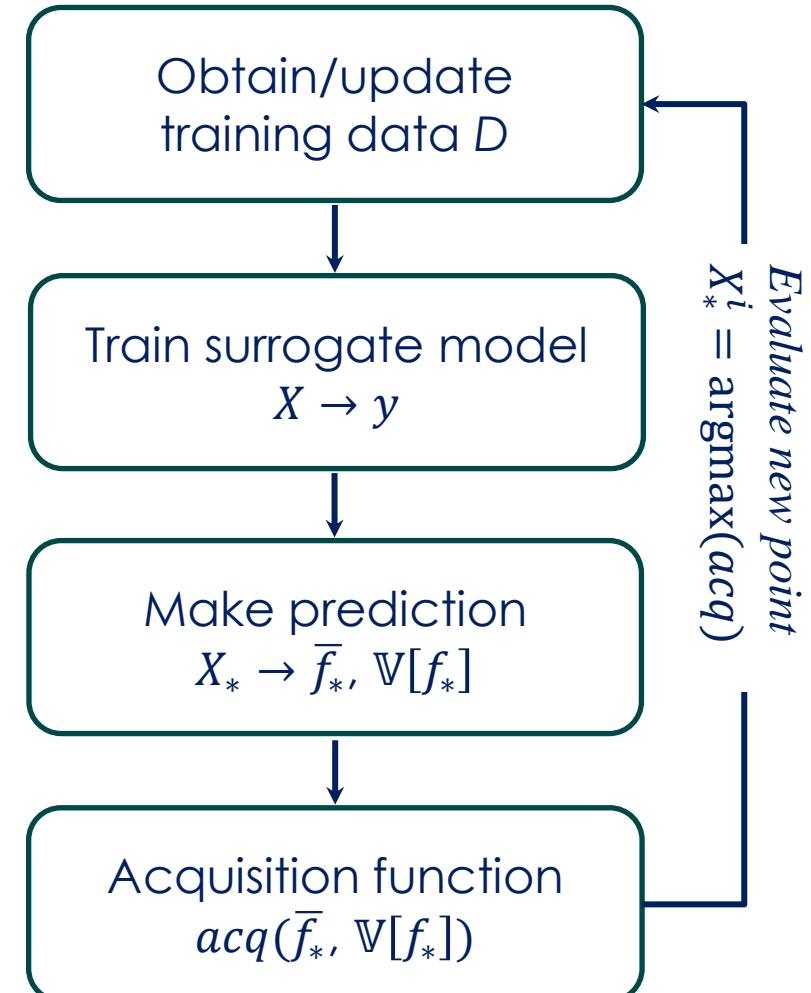
Or we can use simple Gaussian Process-based Bayesian Optimization to do so

# Bayesian Optimization!

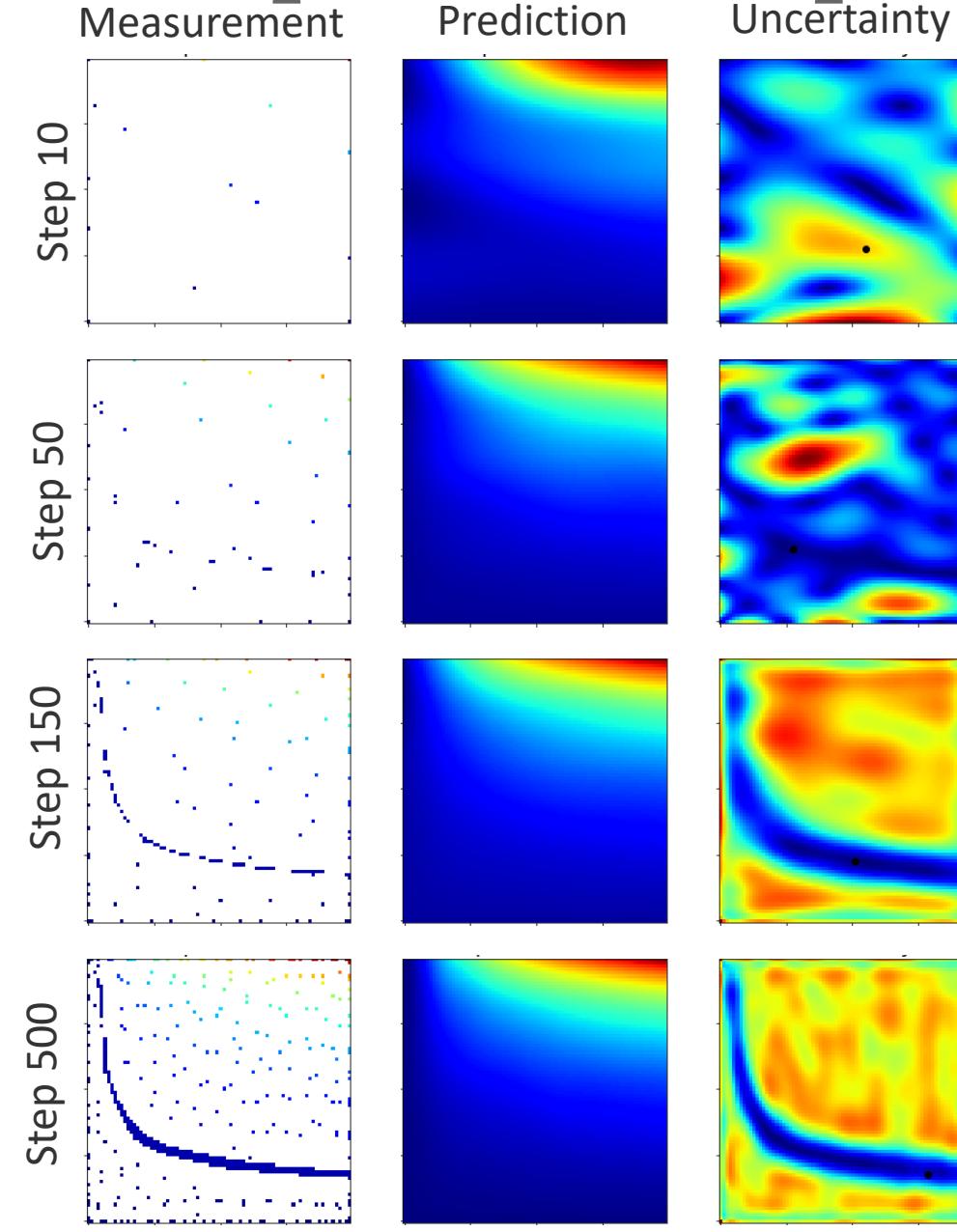


N. de Freitas et al., Taking the Human Out of the Loop: A Review of Bayesian Optimization ,  
Proceedings of the IEEE 104, 148 (2015)

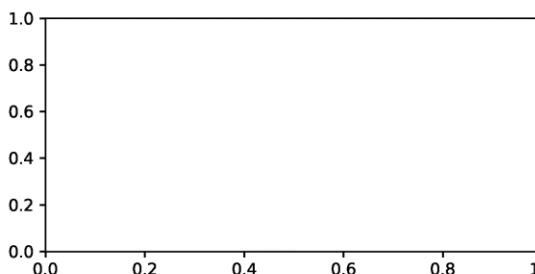
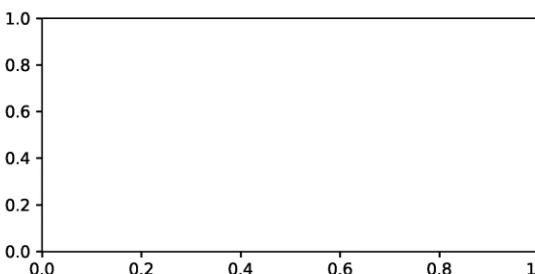
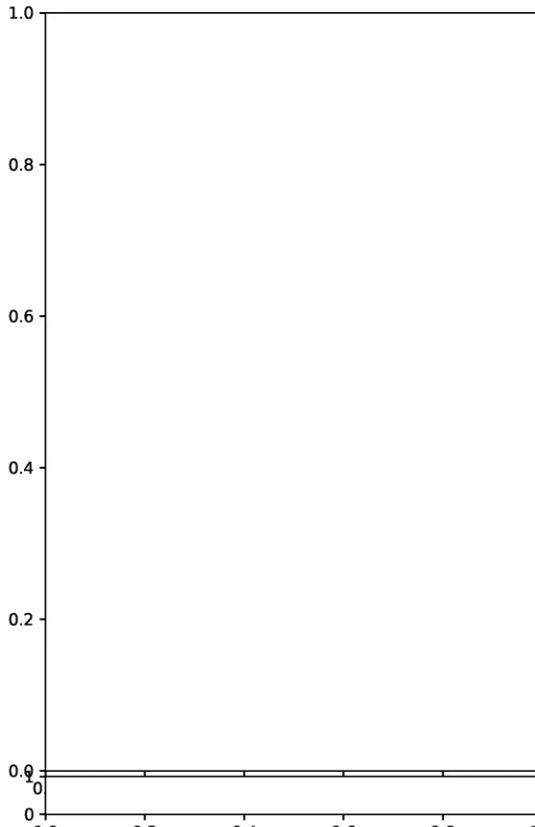
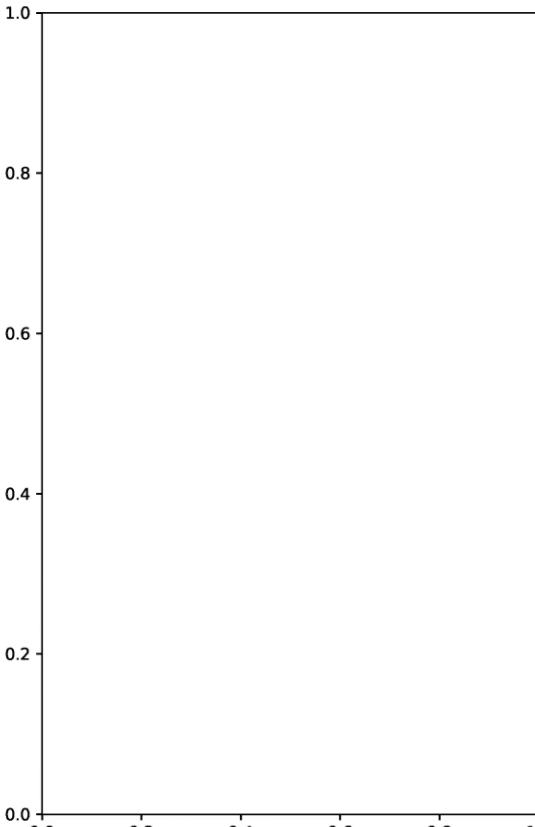
$X, y$ : (sparse) Training data  
 $X_*$ : New (not yet evaluated) points



# BO exploration of parameter space

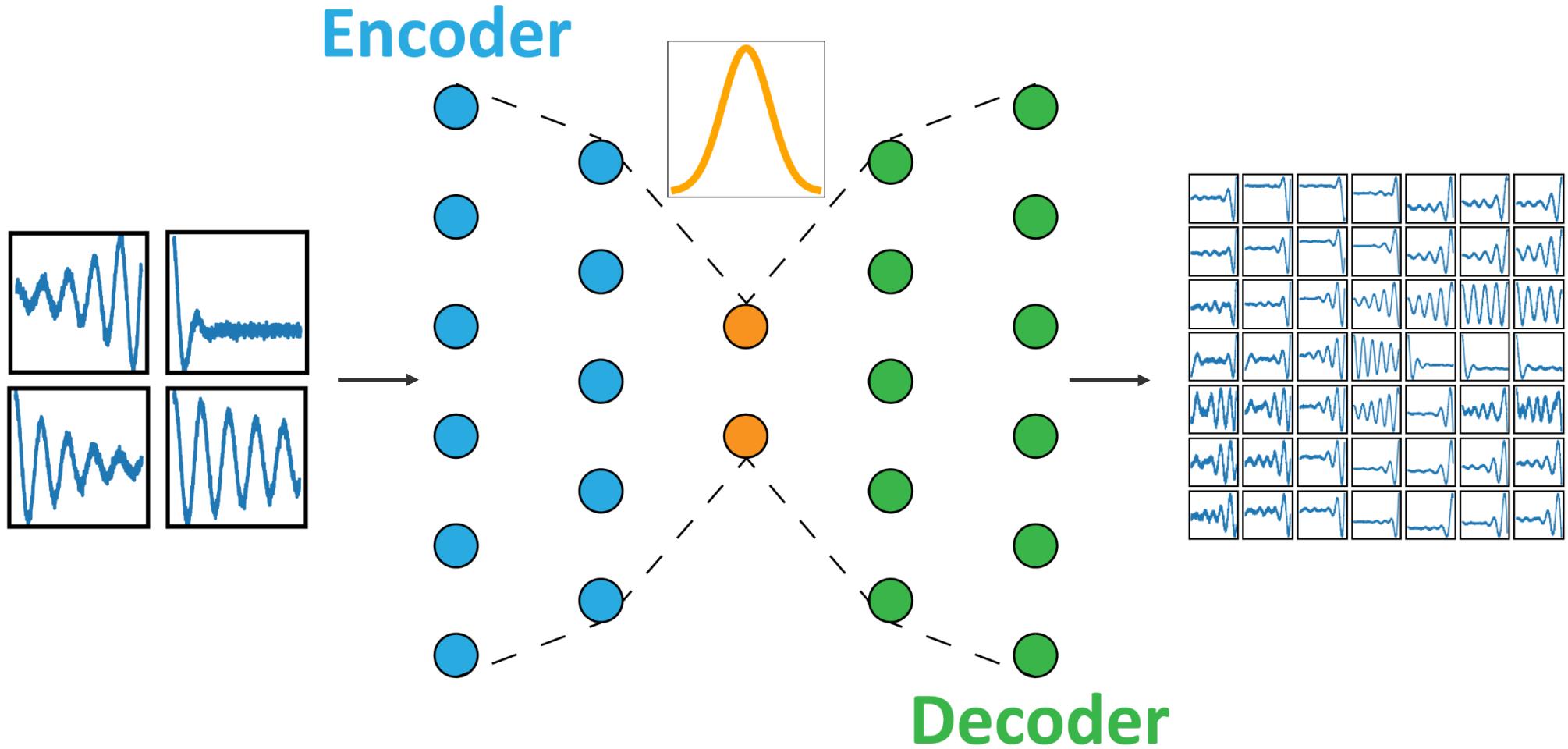


# But what about trajectories?



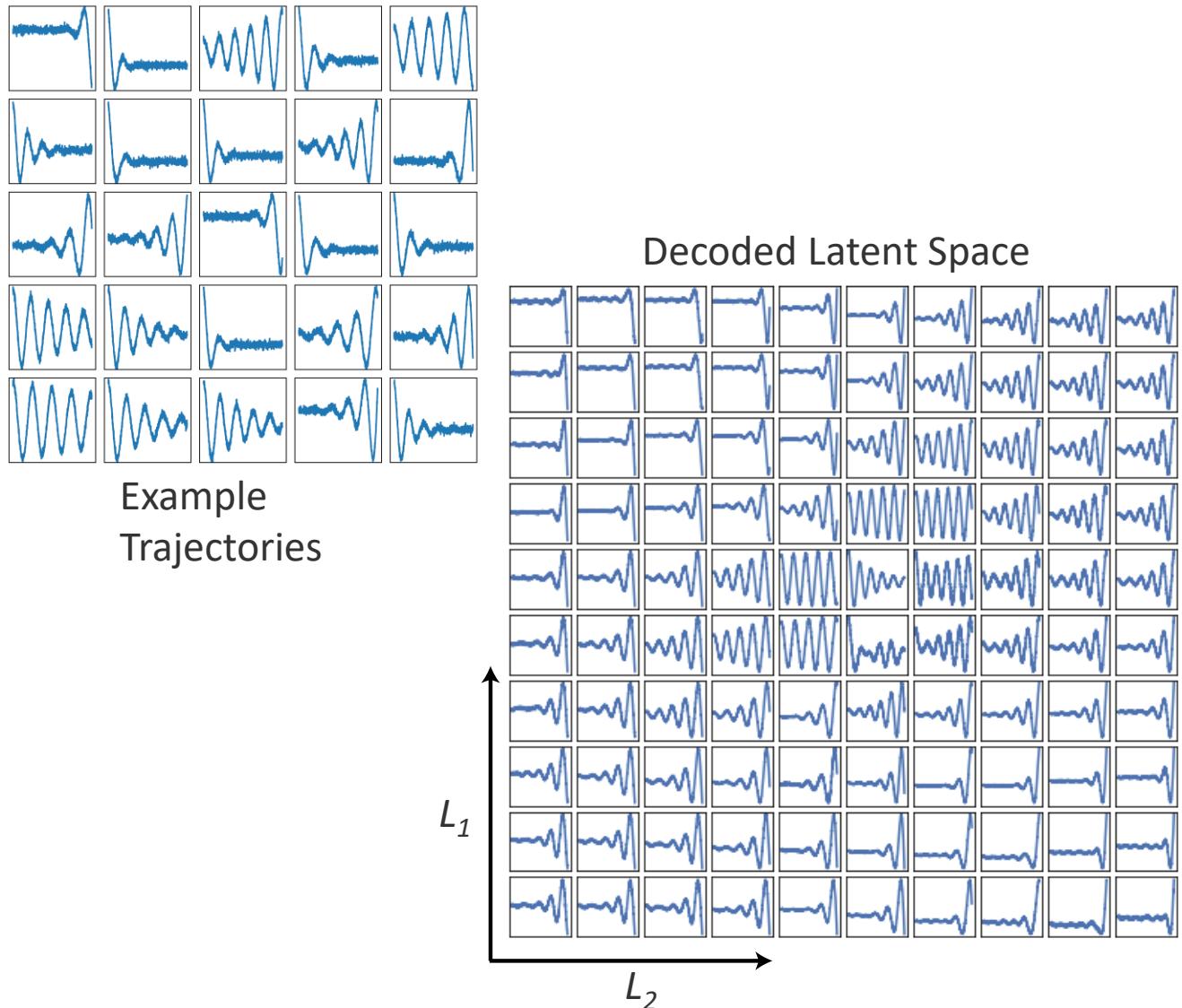
- The model has large number of microstates
- The global state depends on history, i.e. dependence of field vs. time
- Can we somehow optimize the chosen global state in the space of possible histories?
- This space is obviously intractable...
- ... however, we are not interested in ALL possible histories. 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?

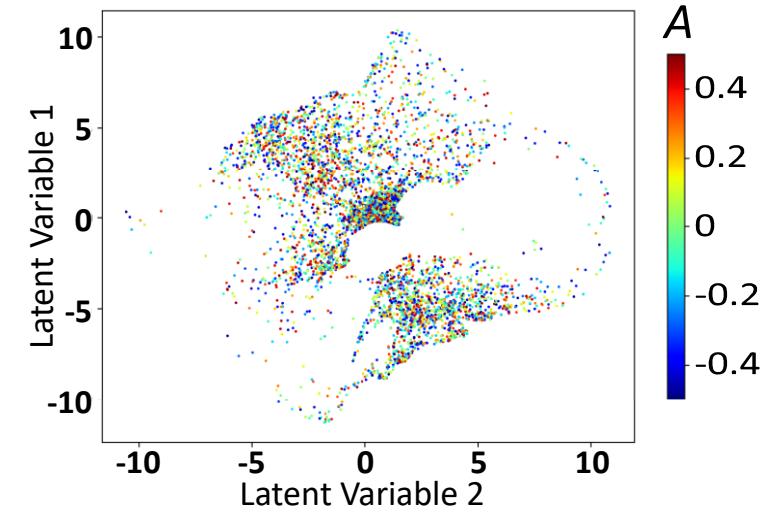
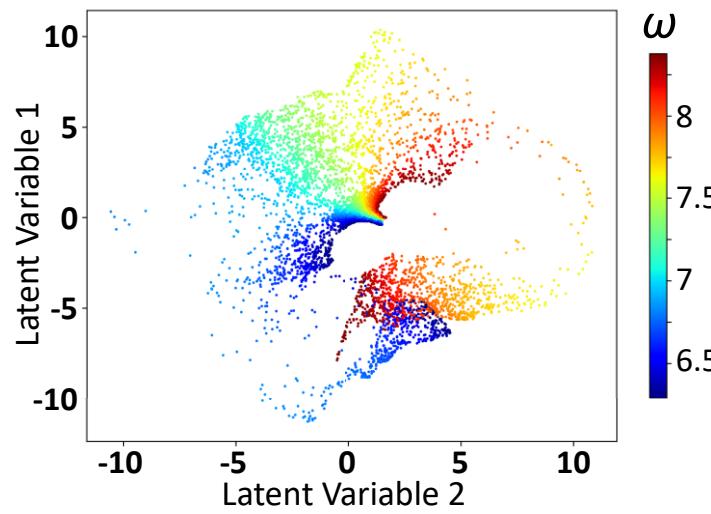
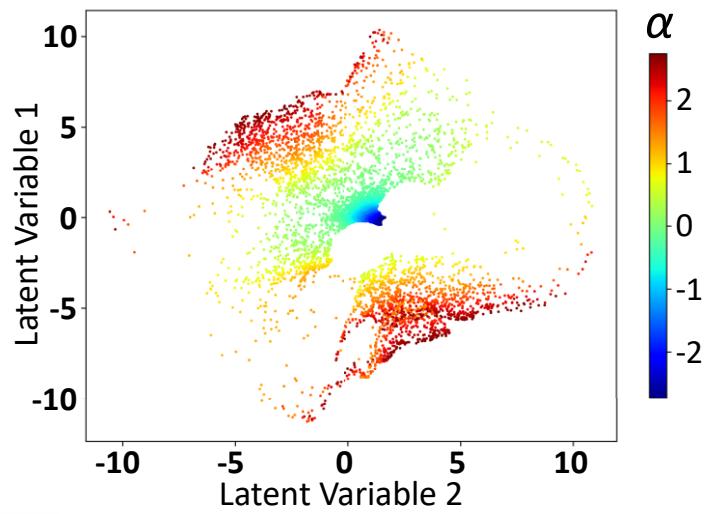


# VAE encoding of domain trajectories

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

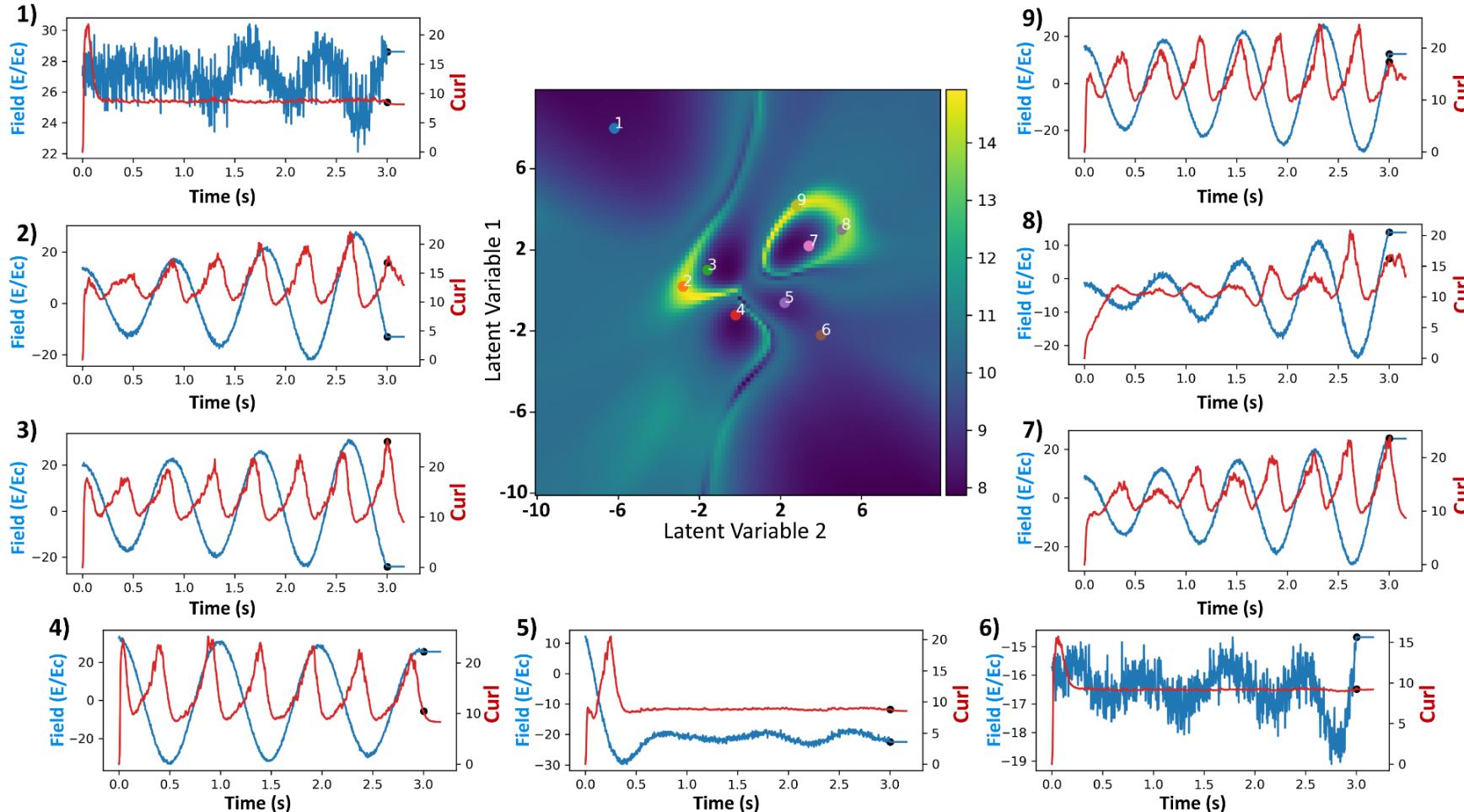


# Latent space distributions



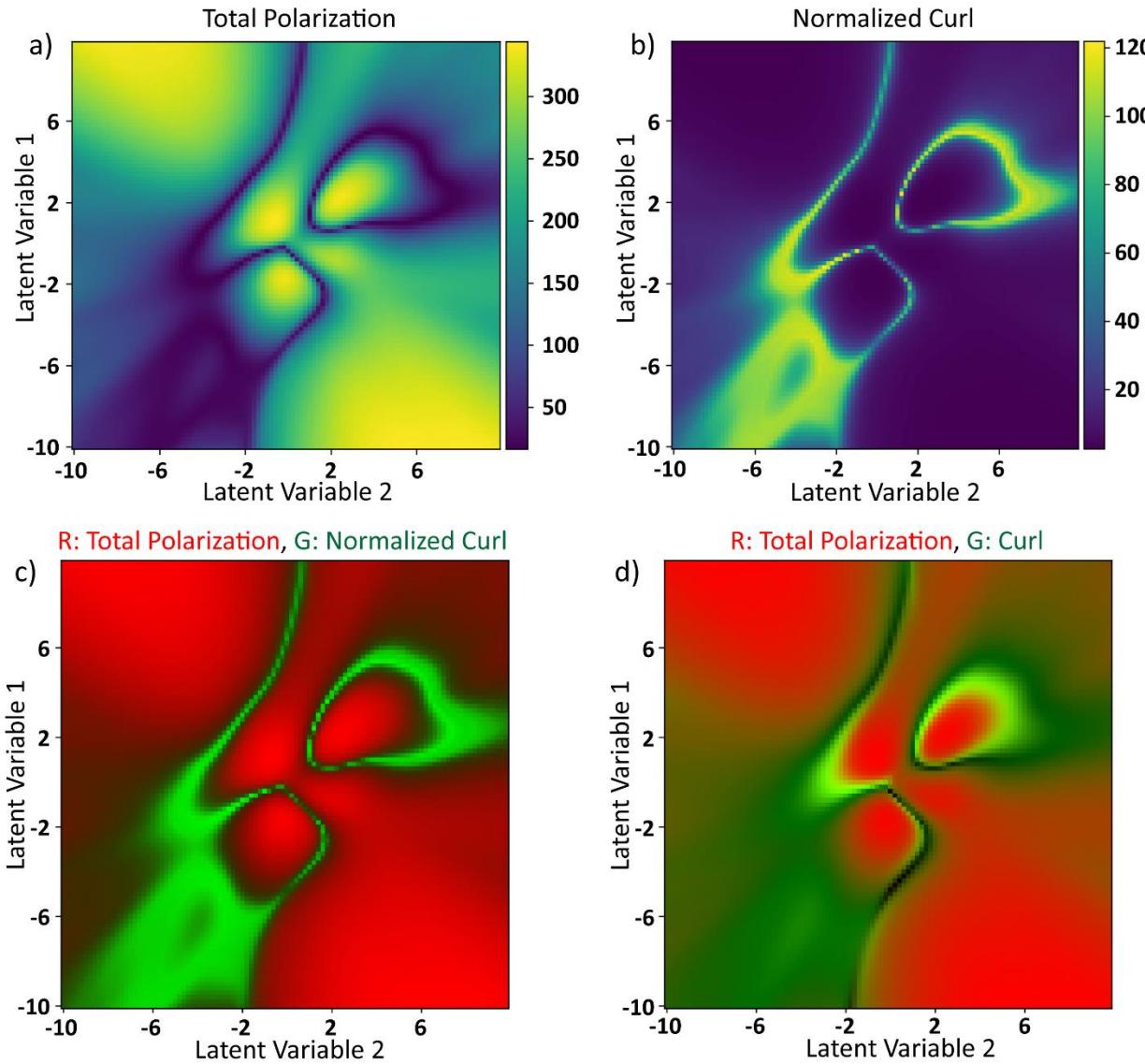
# Ground truth target function

- 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



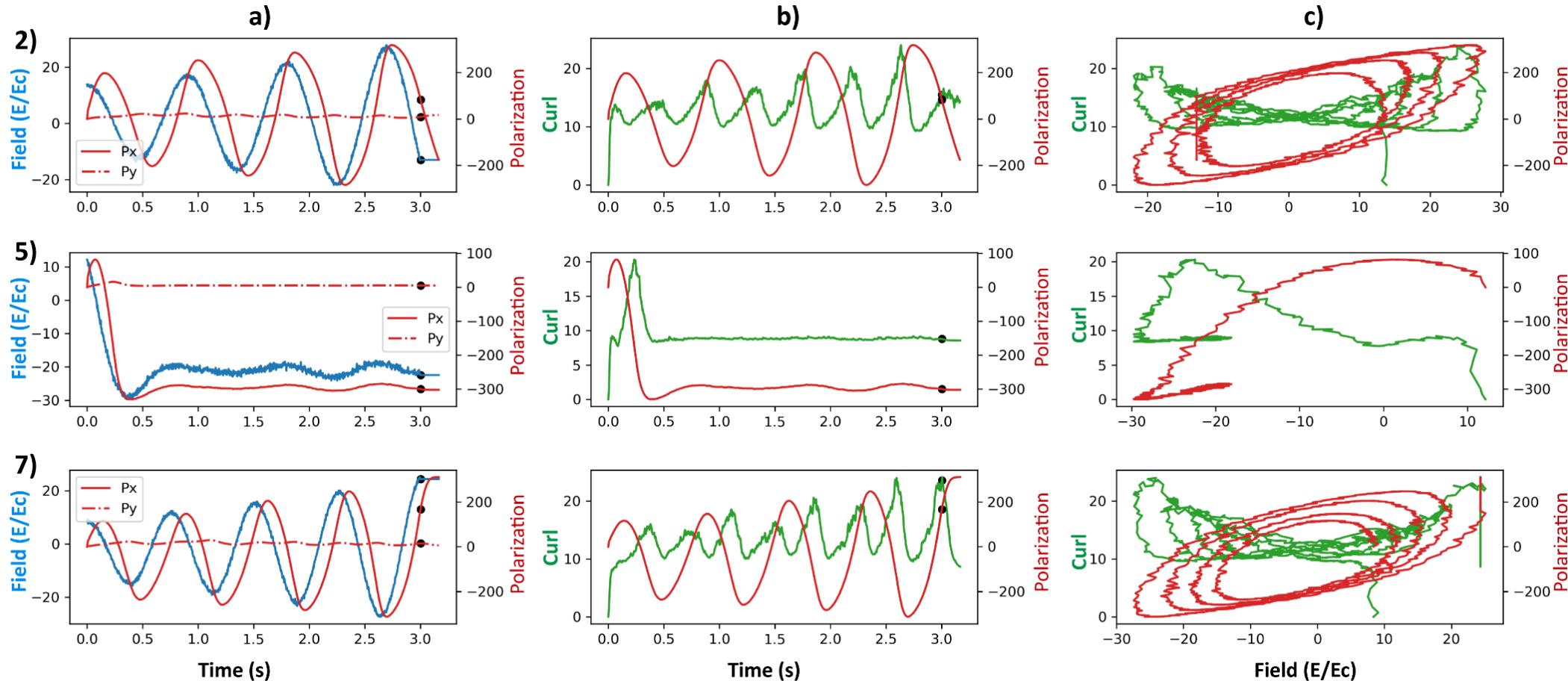
## Normalized Curl

- 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

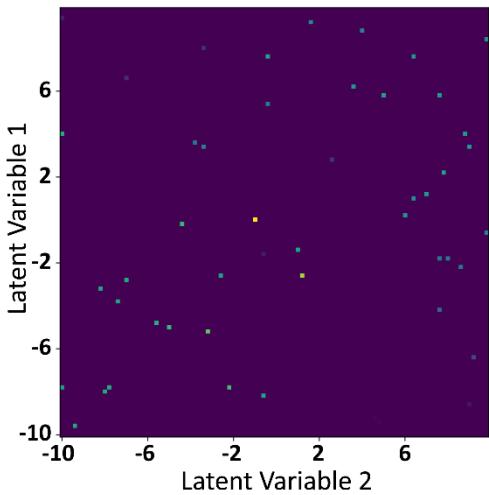
# Exploring the curl surface



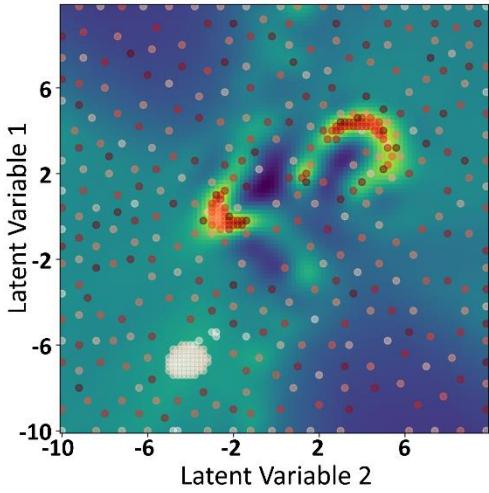
- The system's polarization is at the lowest the coercive field (A state of maximum normalized curl)
- But the curl is also a function of magnitude of the polarization
- Hence, the magnitude of the curl is maximum a few steps after the coercive field where the polarization grows in magnitude just enough 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

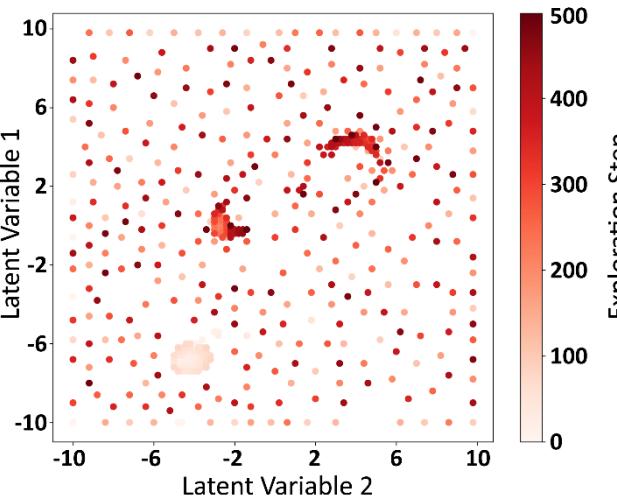
100 initial points



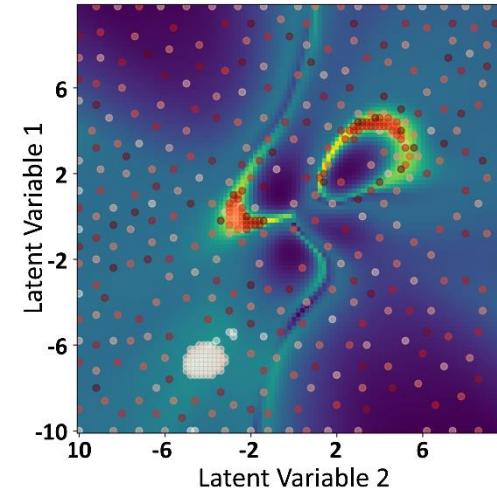
Reconstructed curl surface



Explored points

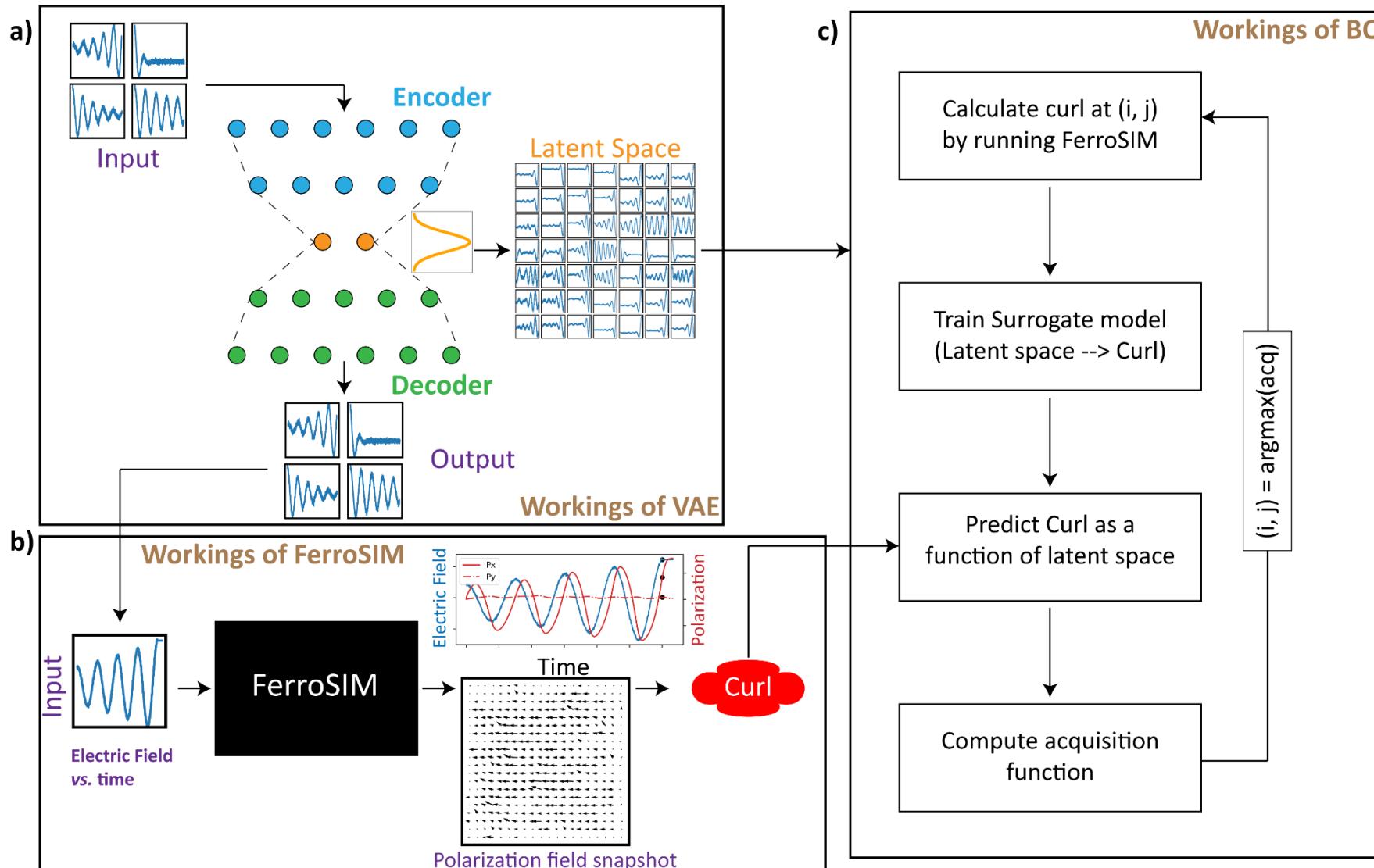


Original curl surface



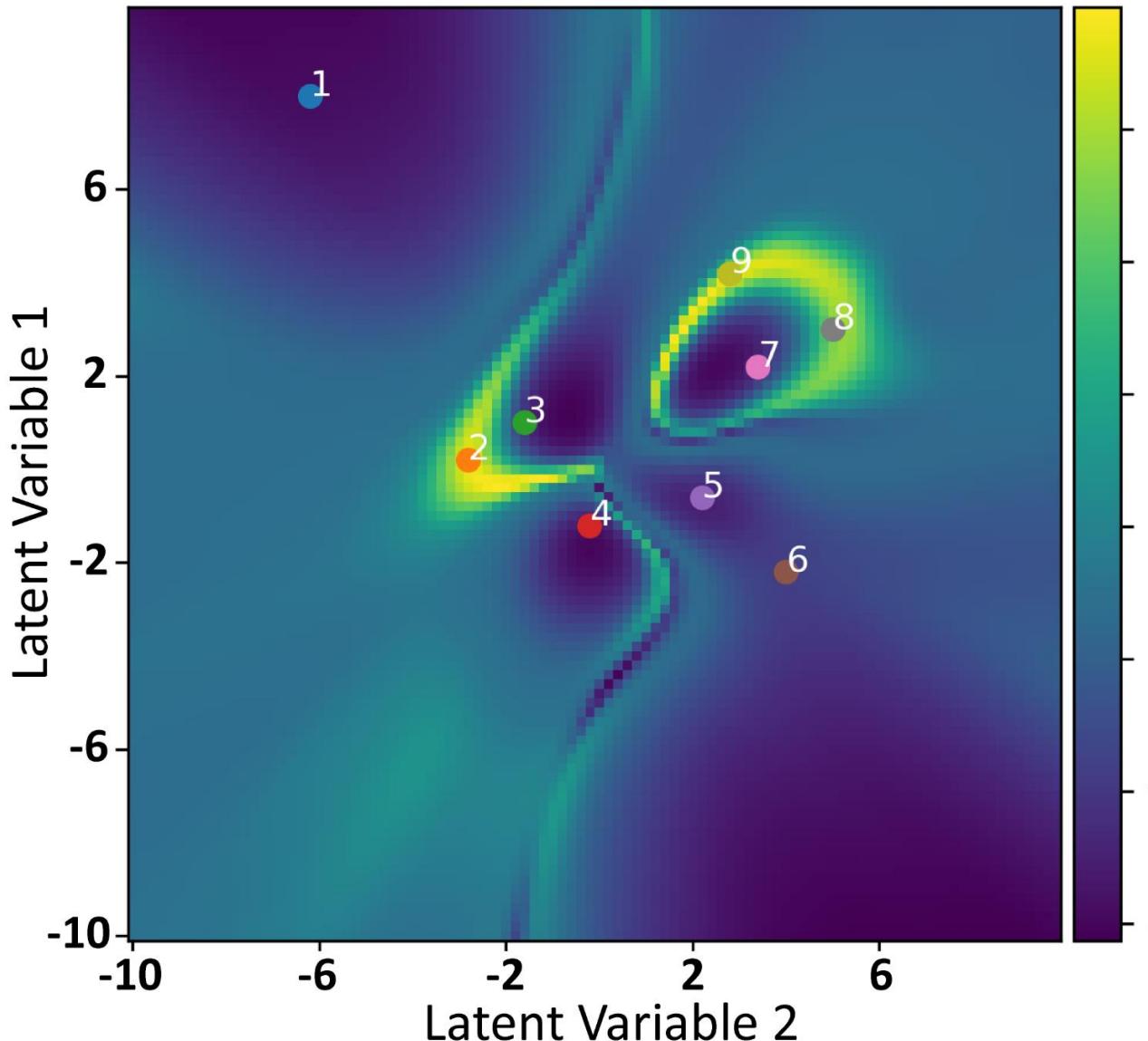
- 100 initialization points and the BO explored the latent space for the next 500 points
- Acq function:  $\mu + 10\sigma$
- So, at the end BO only explored a total of 600 points out of 10,000 points the latent space is divided into
- Caveat: we had to tune the Acq with the ground truth data known

# Putting everything together



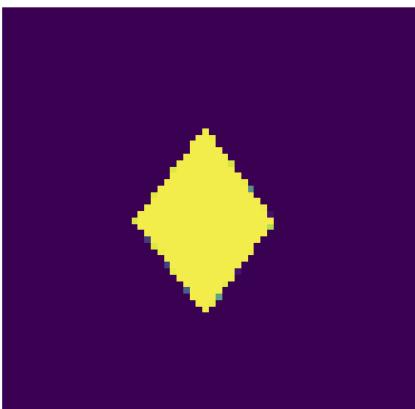
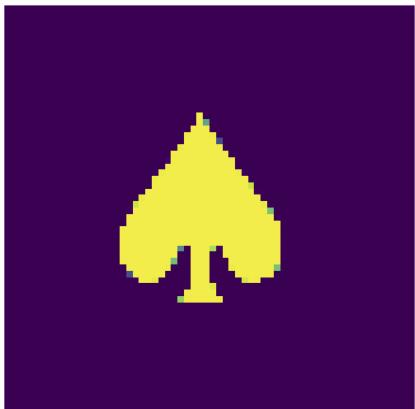
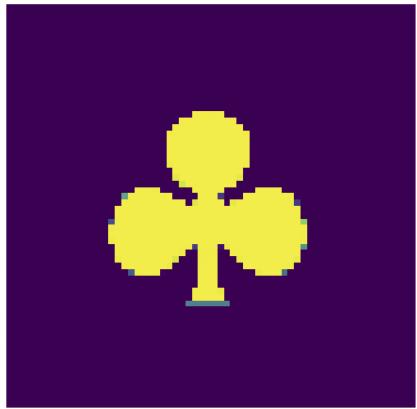
Same approaches are used for molecular discovery, polymer structure optimization, and biomolecules

# What determines success?

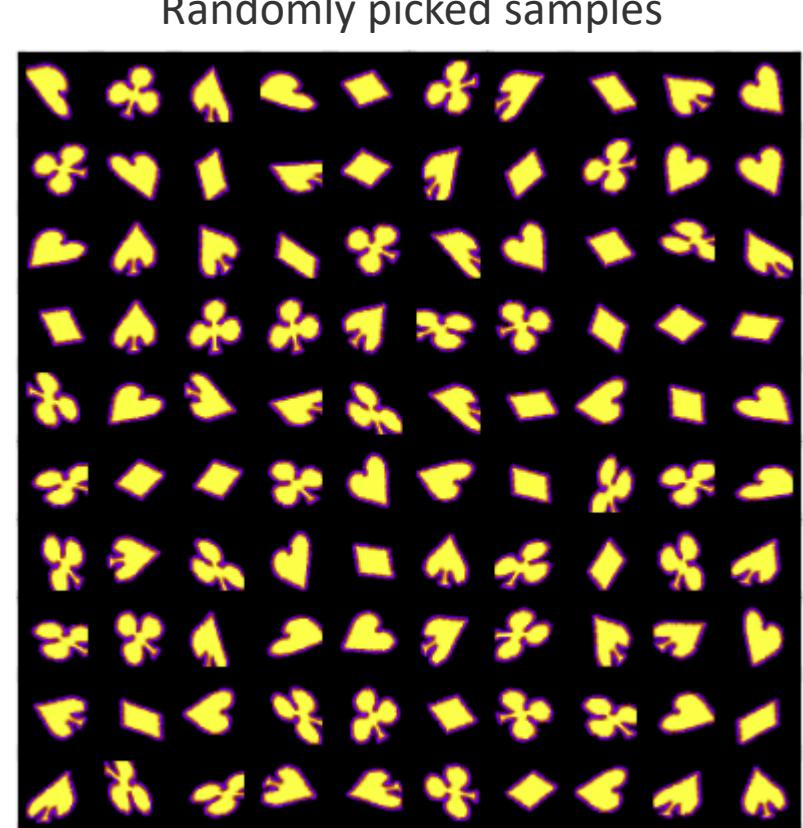


- 14 The success of the BO in the latent space clearly depends on the shape on the manifold that points of interest form.
- 13
- 12 For VAE, the shape of the manifold is determined by the properties of the data only, including
  - (a) how strong correlations in data reflect in correlation in properties and
  - (b) weight of the “good” trajectories
- 11
- 10
- 9
- 8

# Card data set

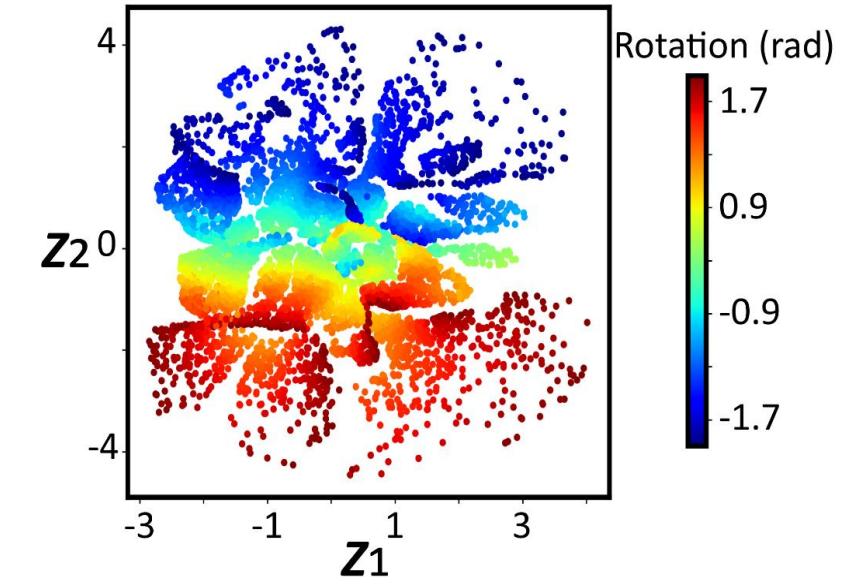
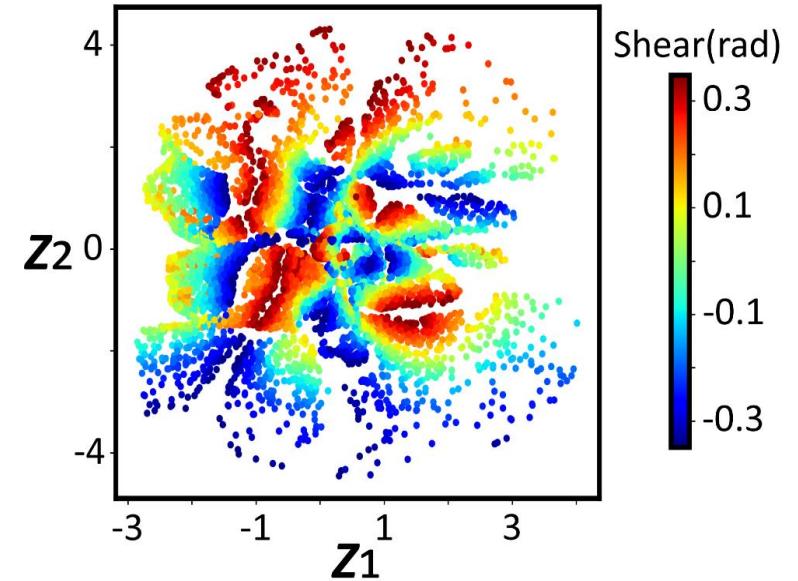
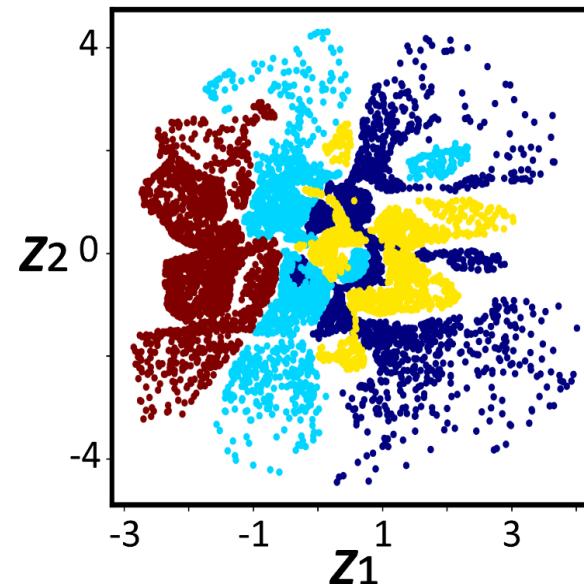
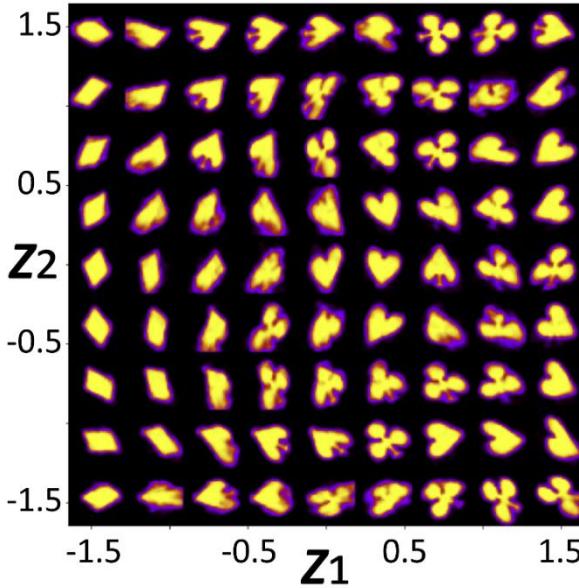


Rotations:  
[-120°, 120°]  
Shear:  
[-20°, 20°]

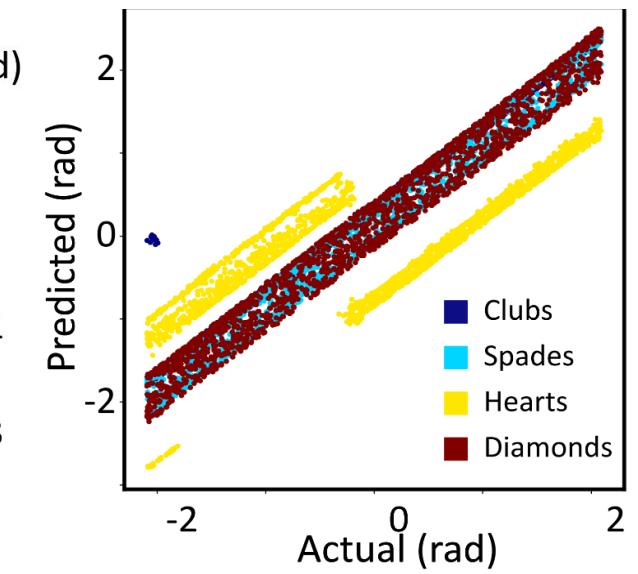
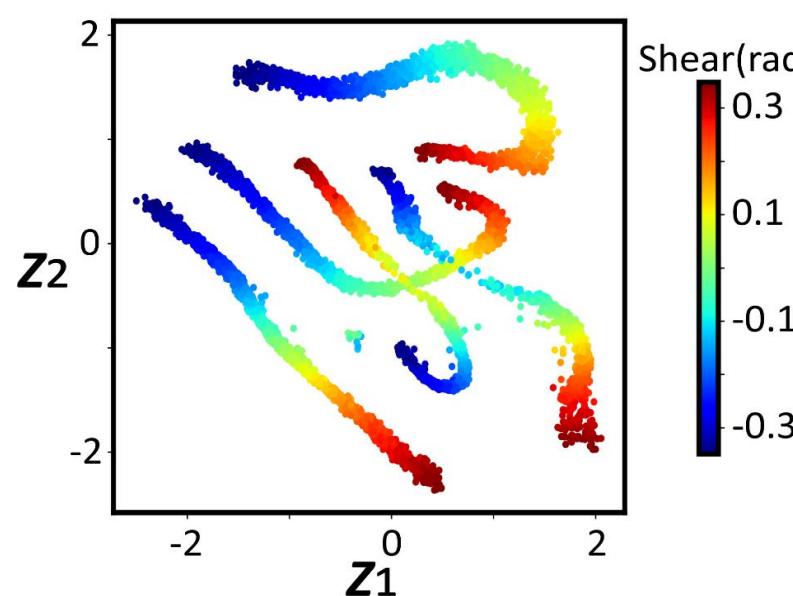
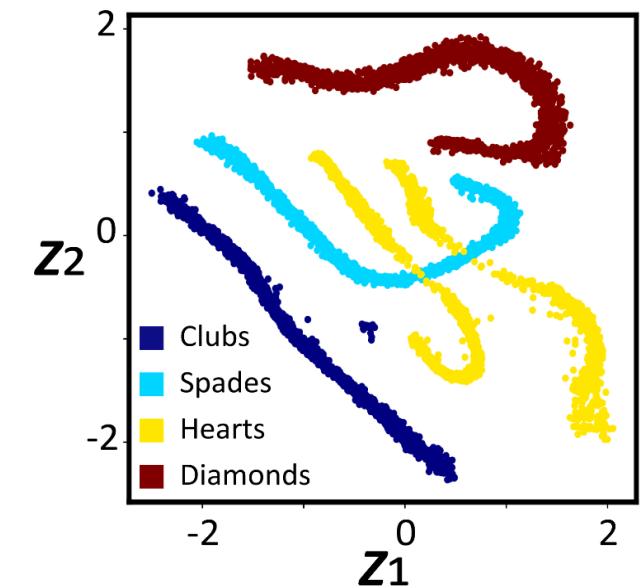
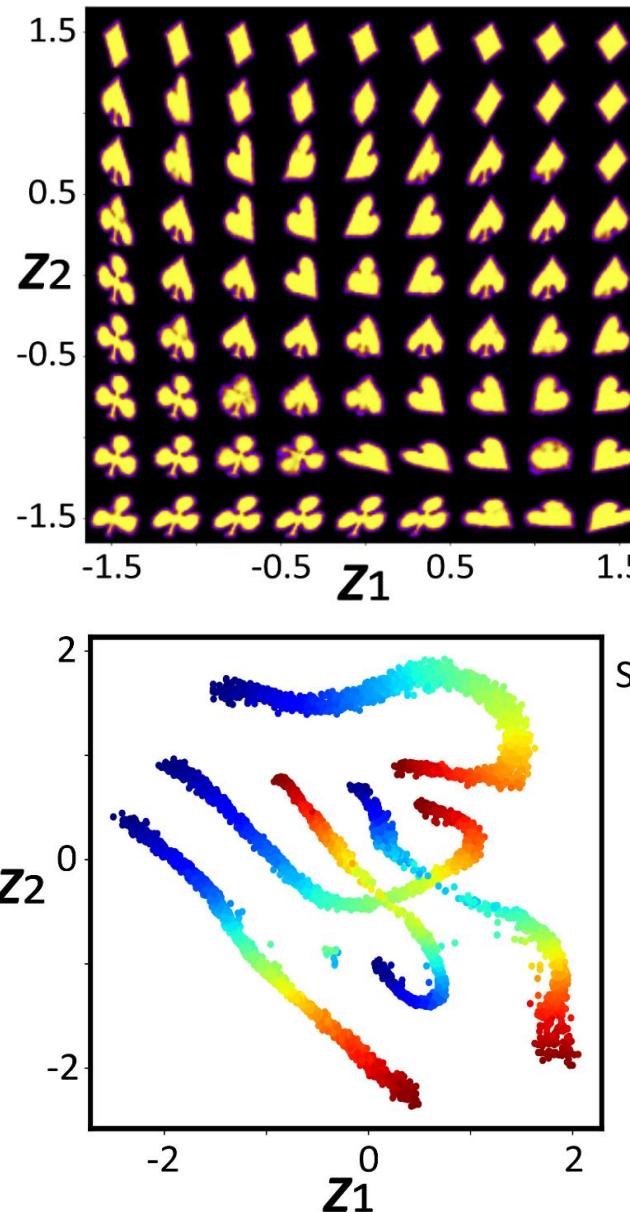


# VAE on Cards

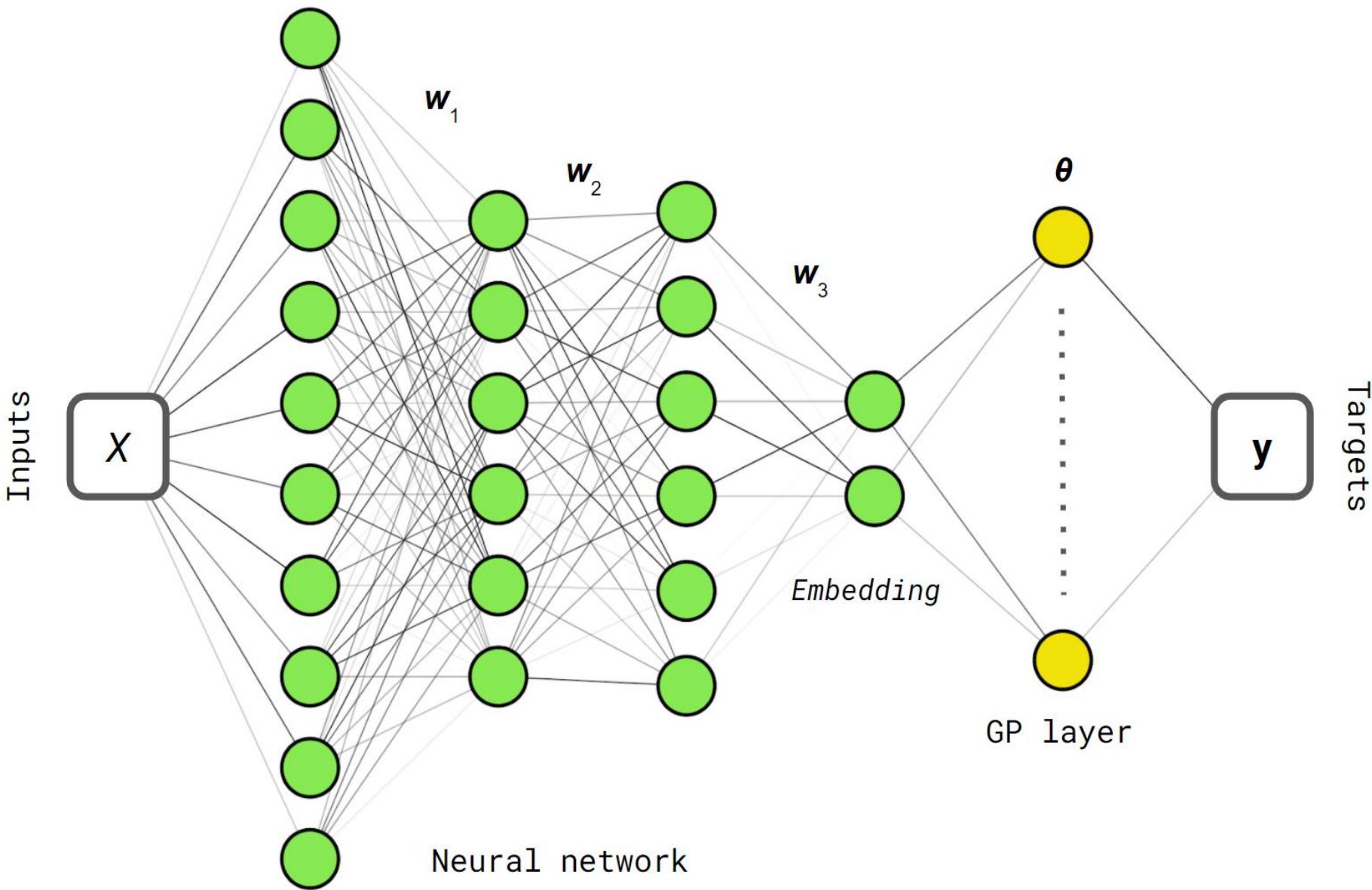
- Clubs
- Spades
- Hearts
- Diamonds



# rVAE on Cards

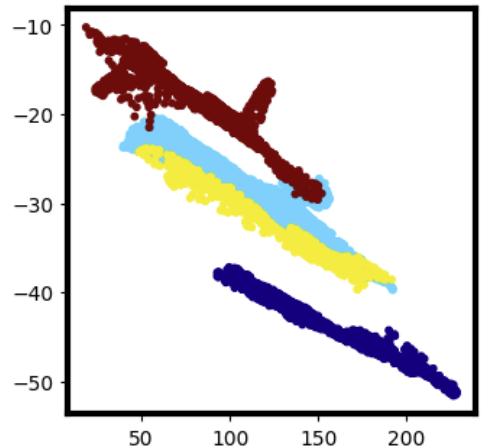


# Deep Kernel Learning

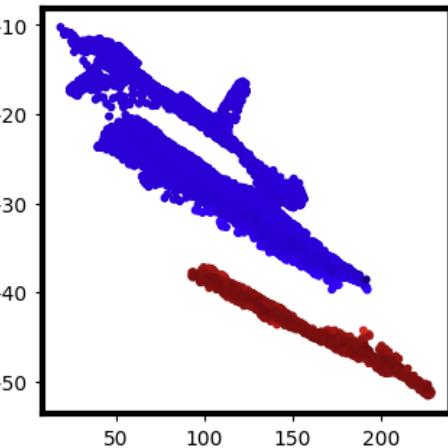


# DKL to predict labels

Clubs

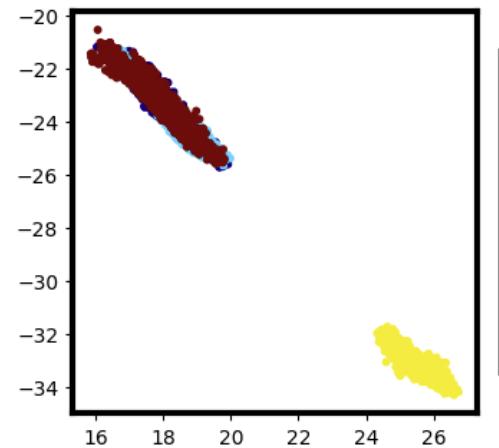


Actual\_Labels

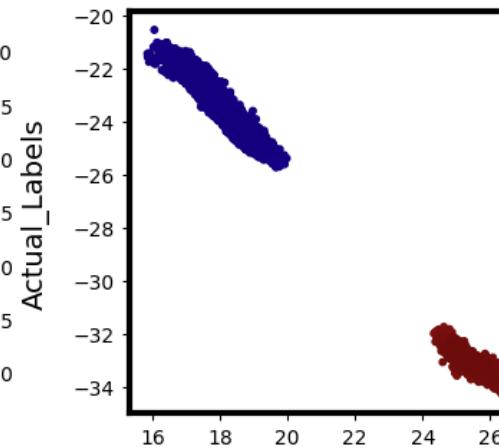


Predicted\_Labels

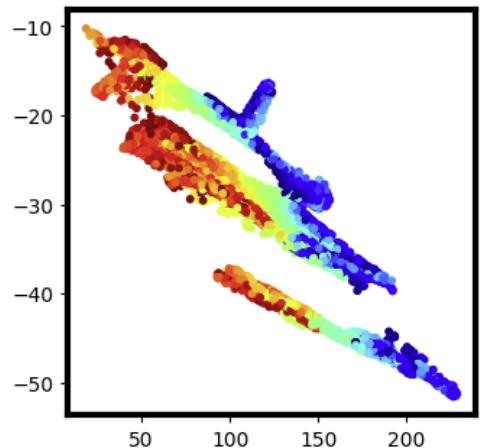
Hearts



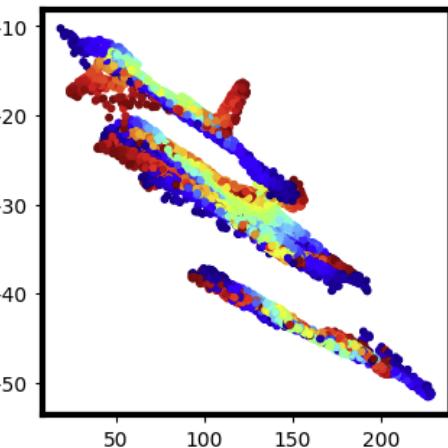
Actual\_Labels



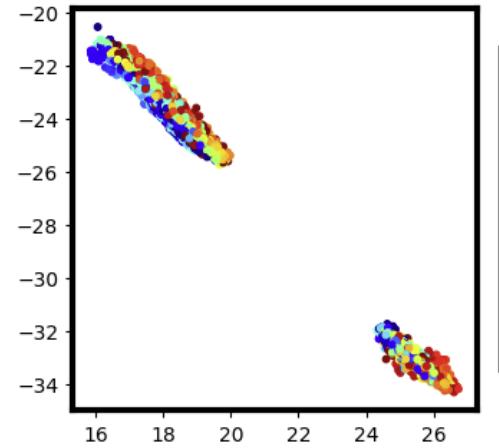
Predicted\_Labels



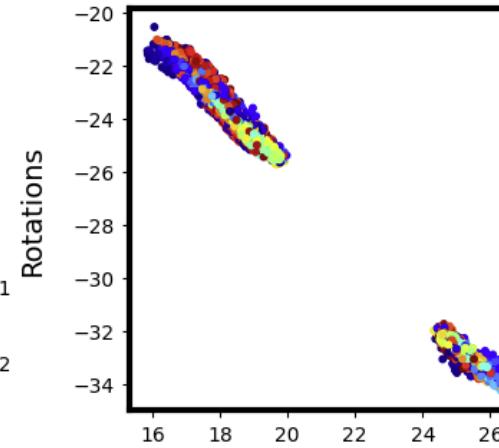
Rotations



Shear



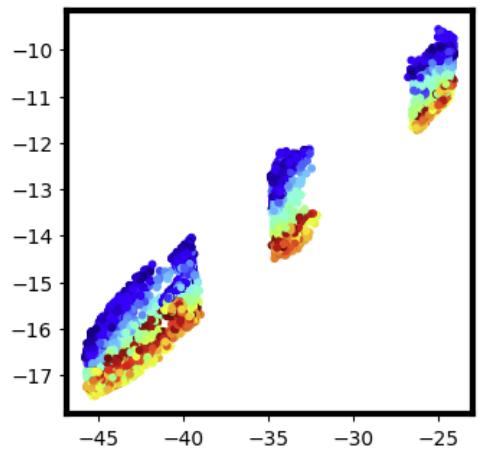
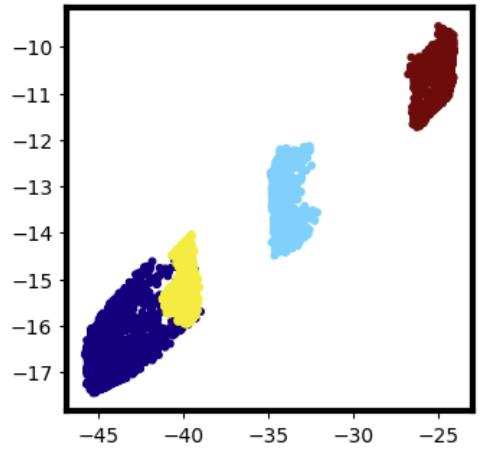
Actual\_Labels



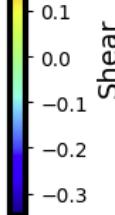
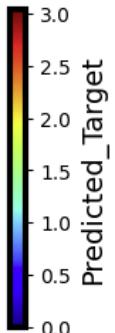
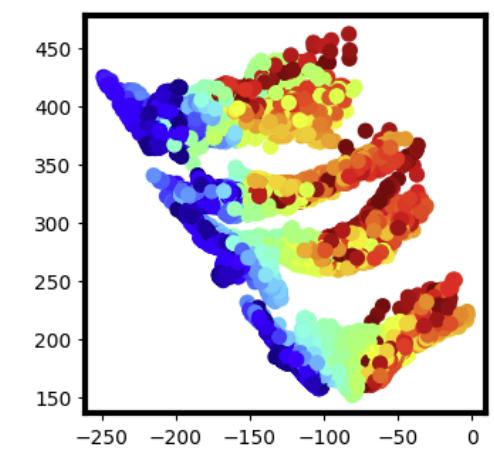
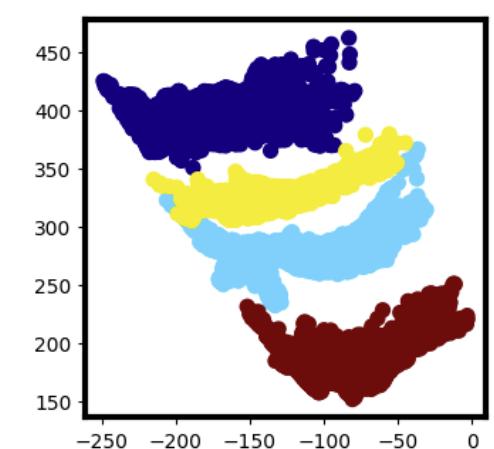
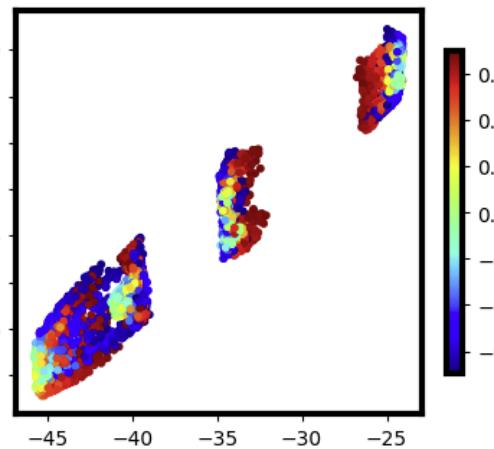
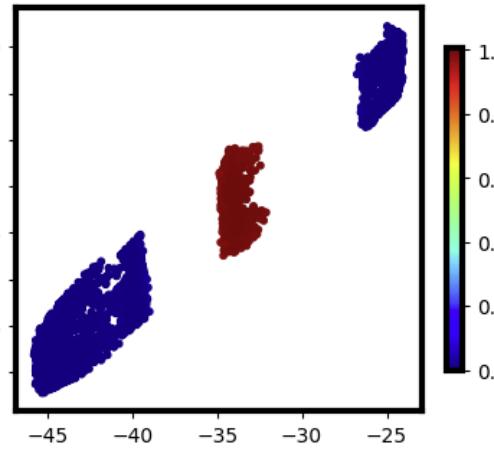
Predicted\_Labels

# DKL to predict labels

Spades



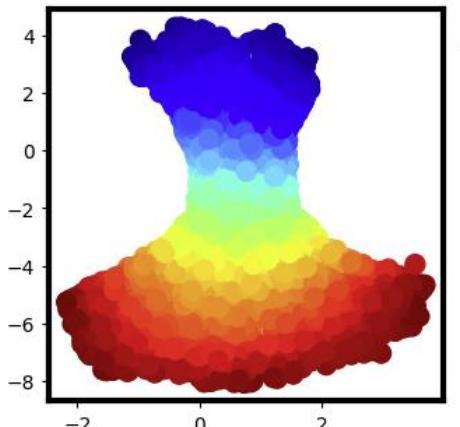
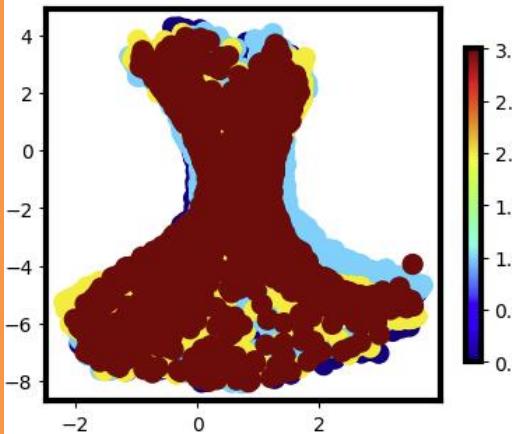
All suits



The DKL clearly forms the manifold based on the label!

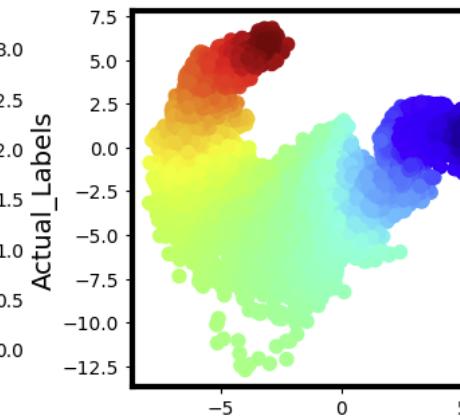
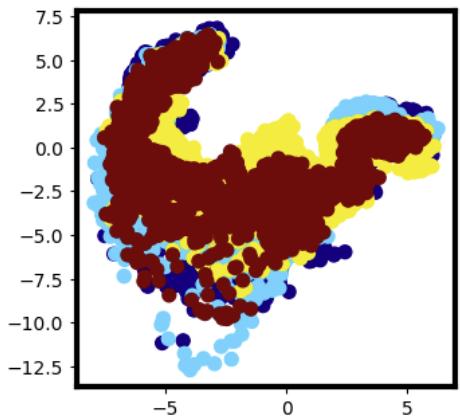
# DKL to predict continuous target function

Shear

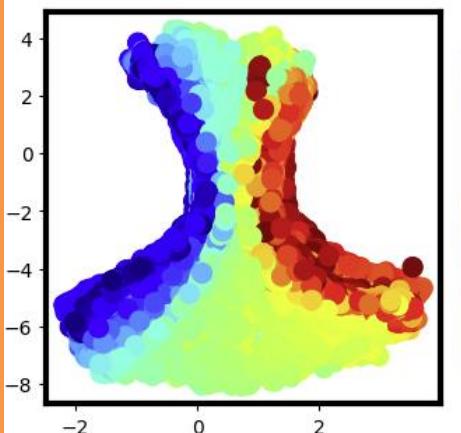


Predicted\_Target

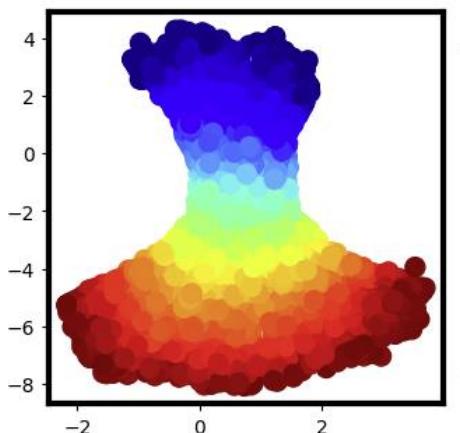
Rotations



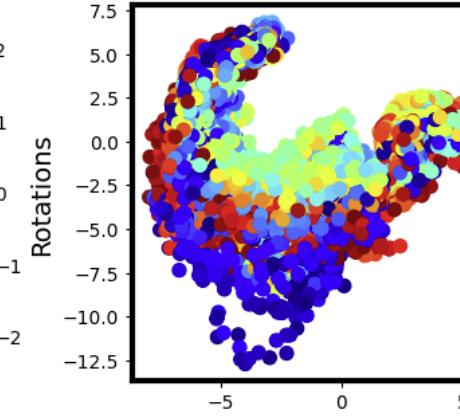
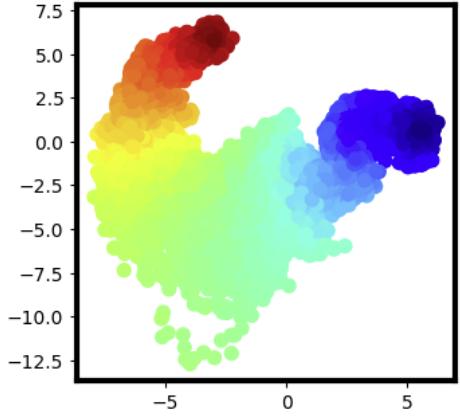
Predicted\_Target



Rotations

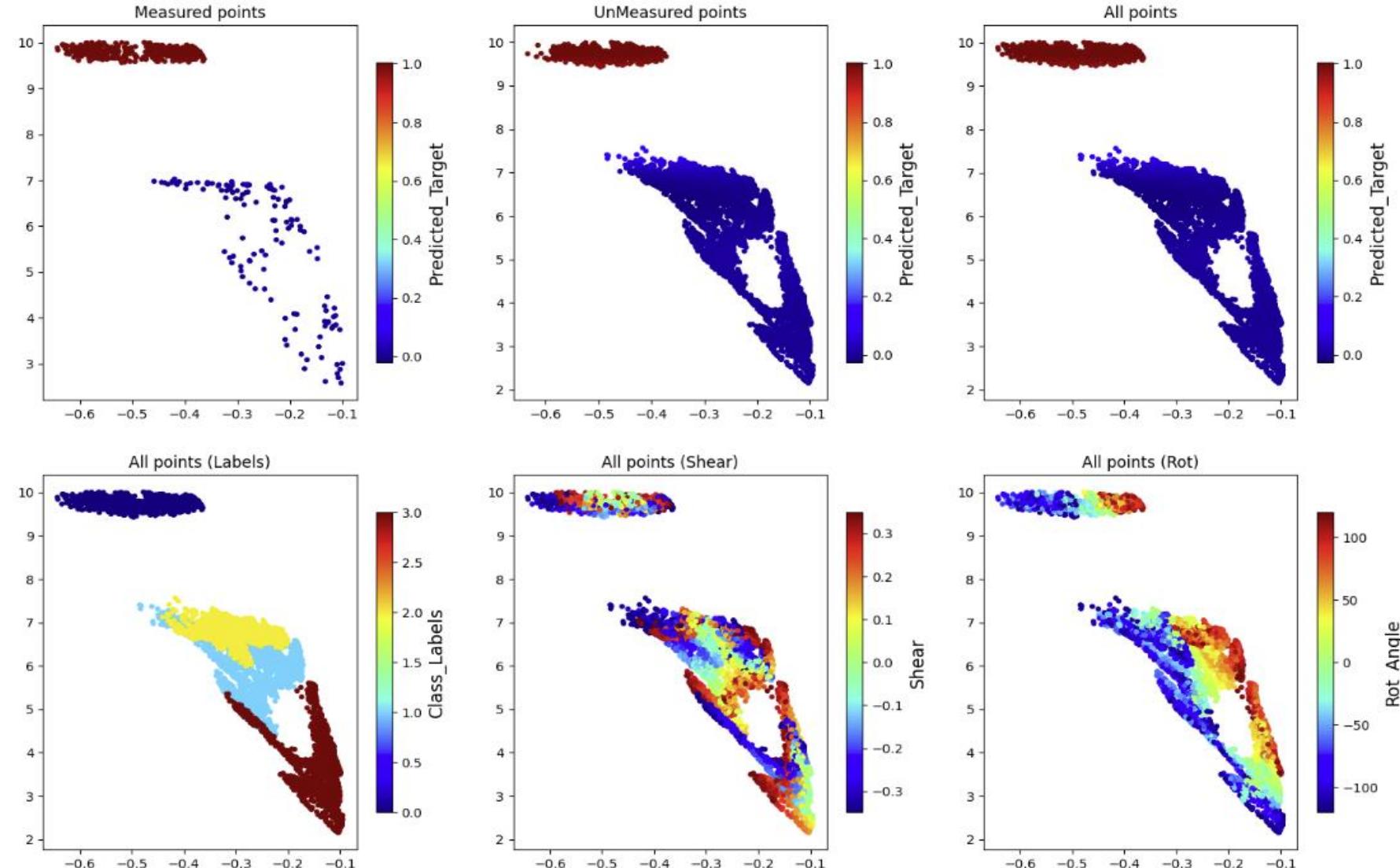


Shear



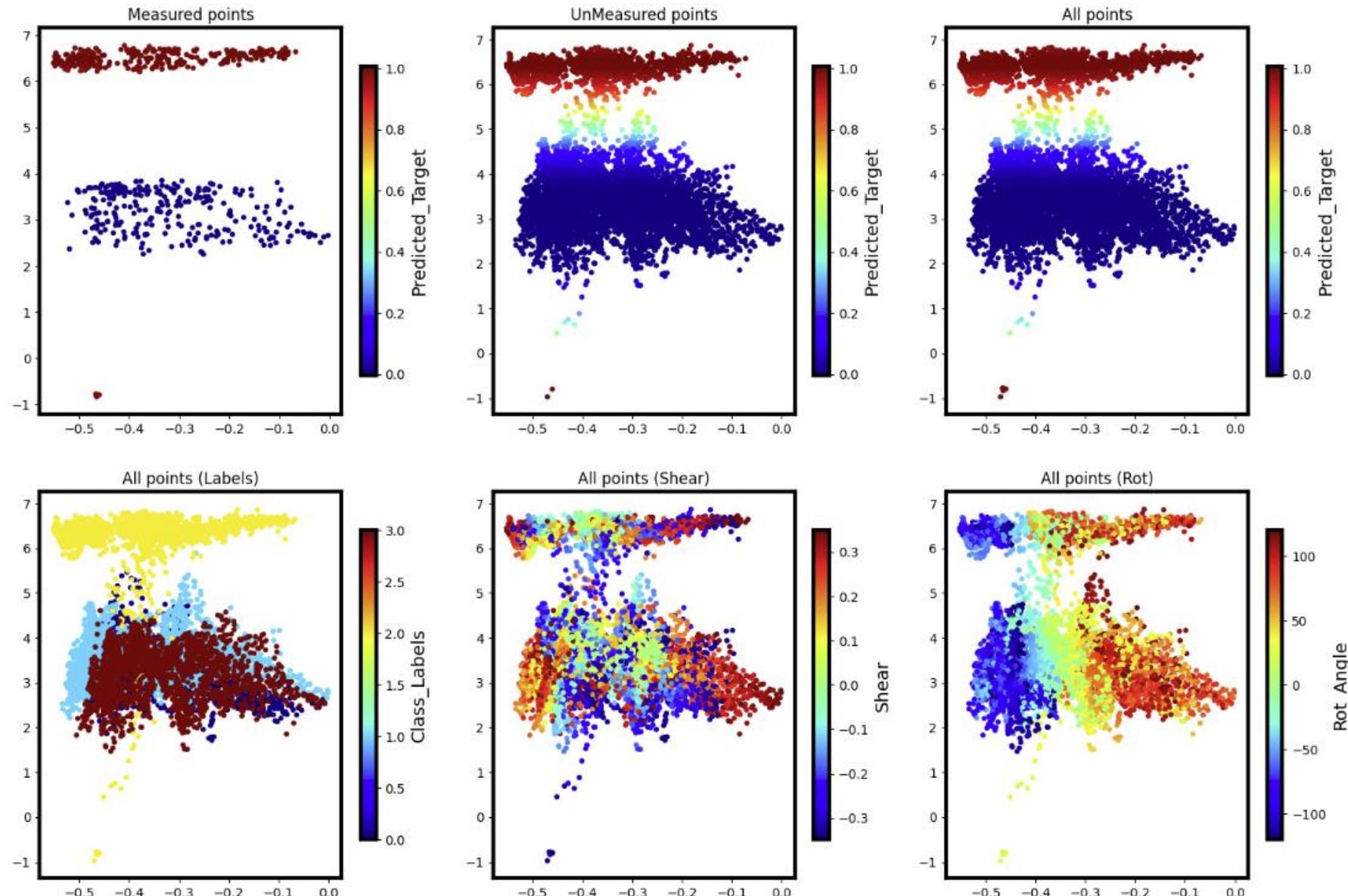
Predicted\_Target

# DKL BO: Active Learning

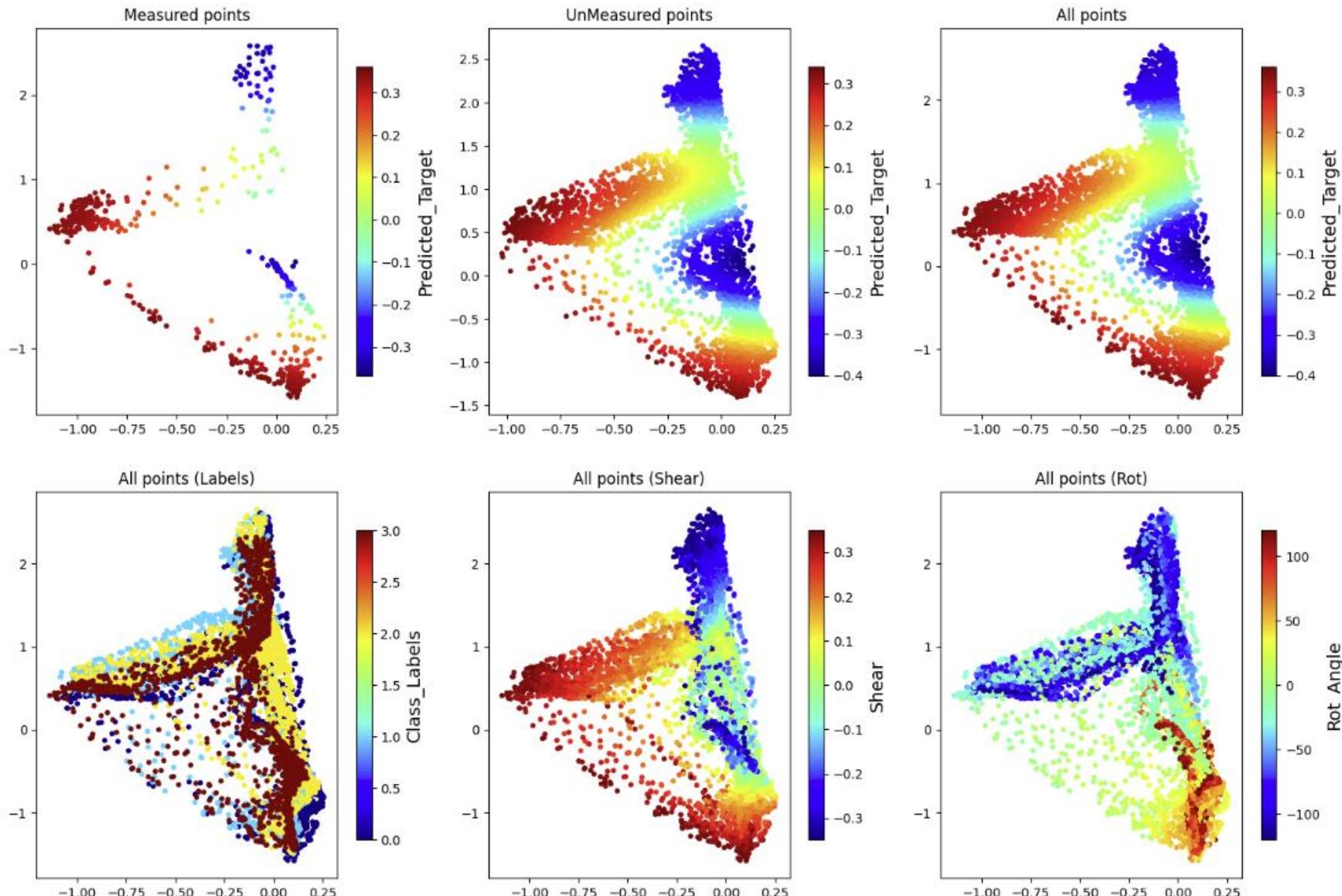


- 100 initialization points and then BO explored 500 points subsequently
- Acquisition function:  $\mu + 10\sigma$

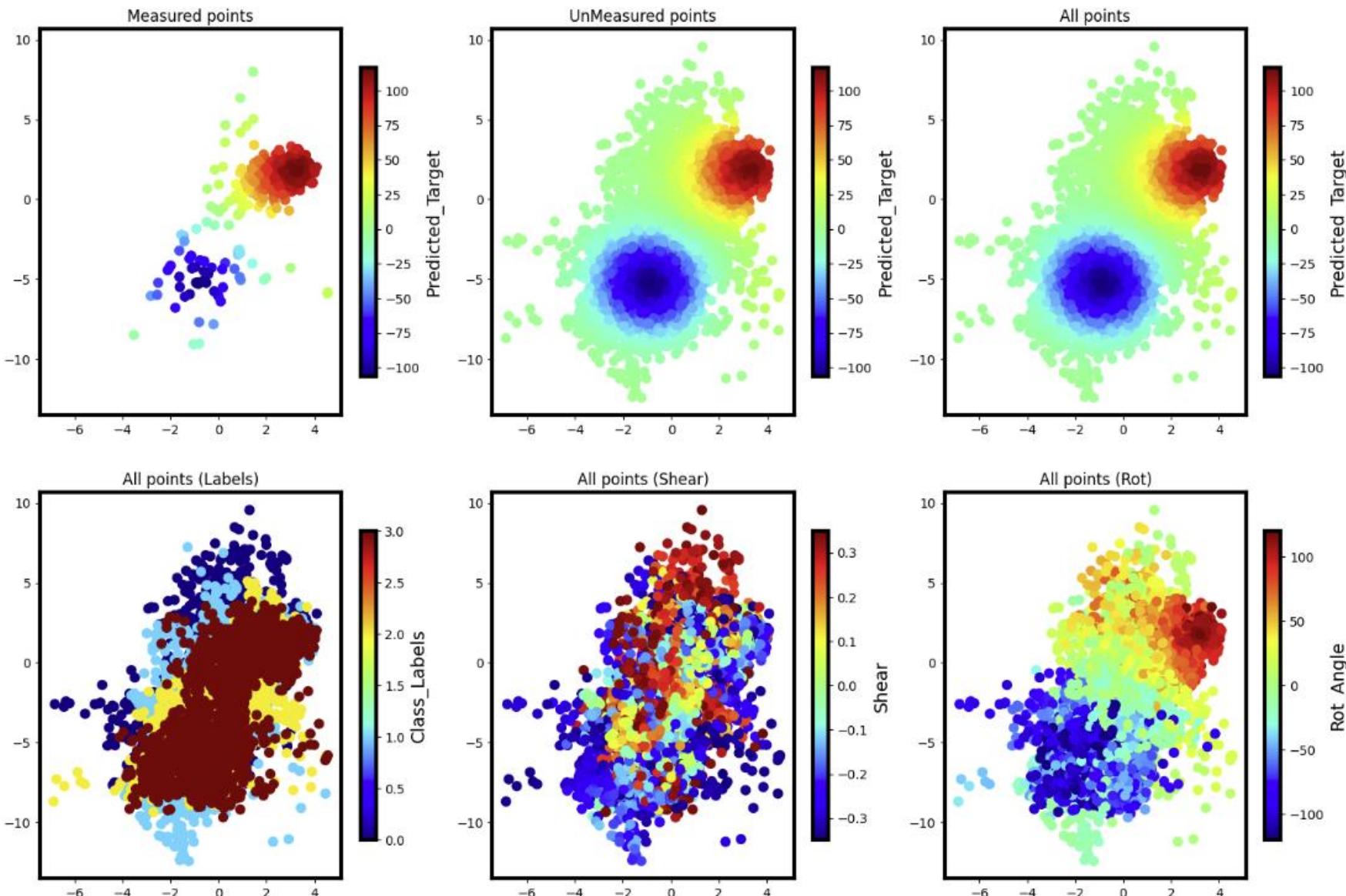
# DKL BO: Hearts



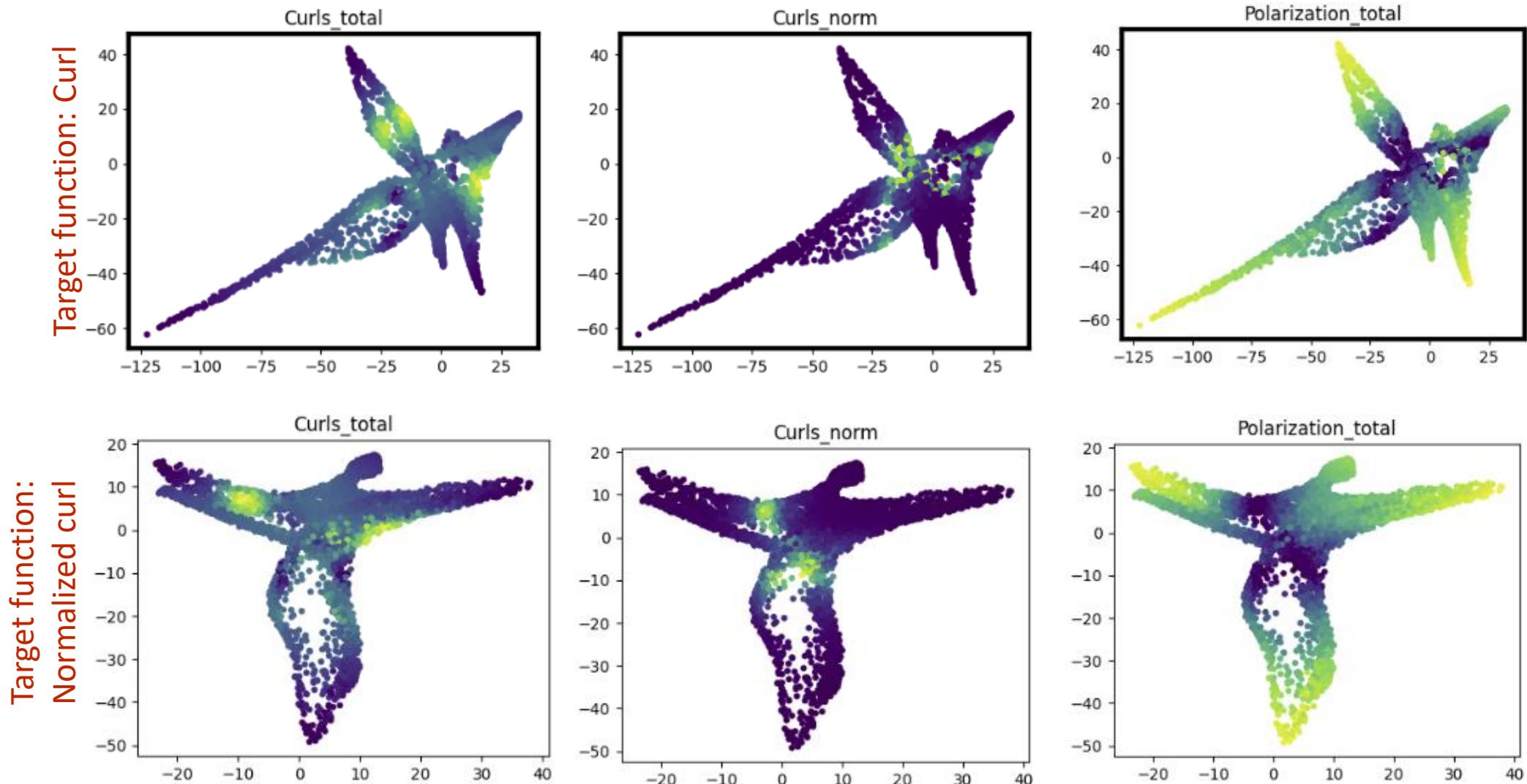
# DKL BO: Shear



# DKL BO: Rotations

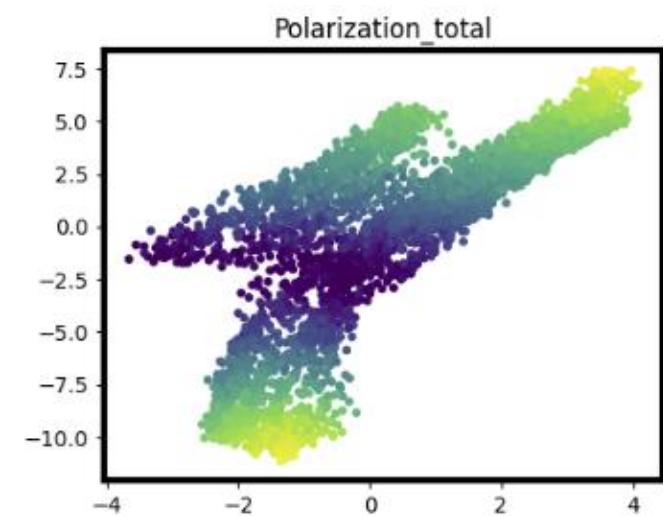
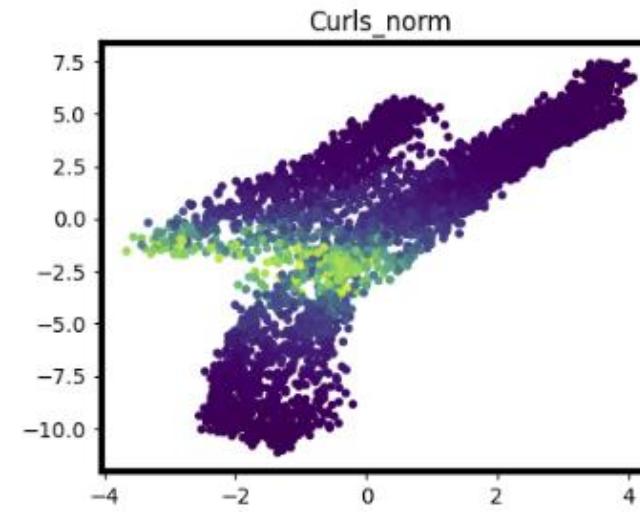
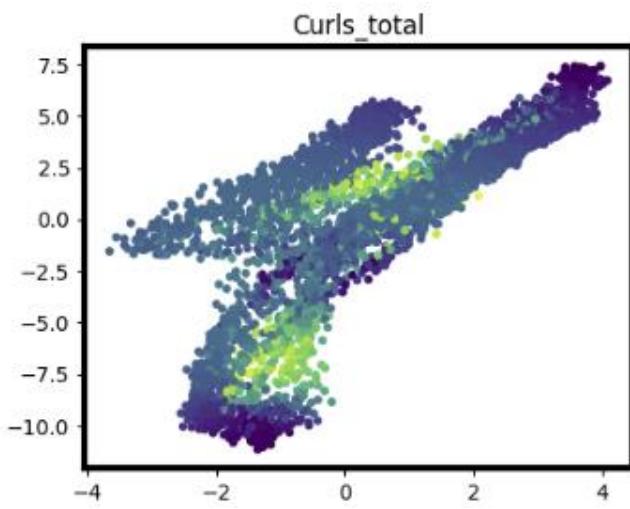


# DKL on FerroSIM: Static

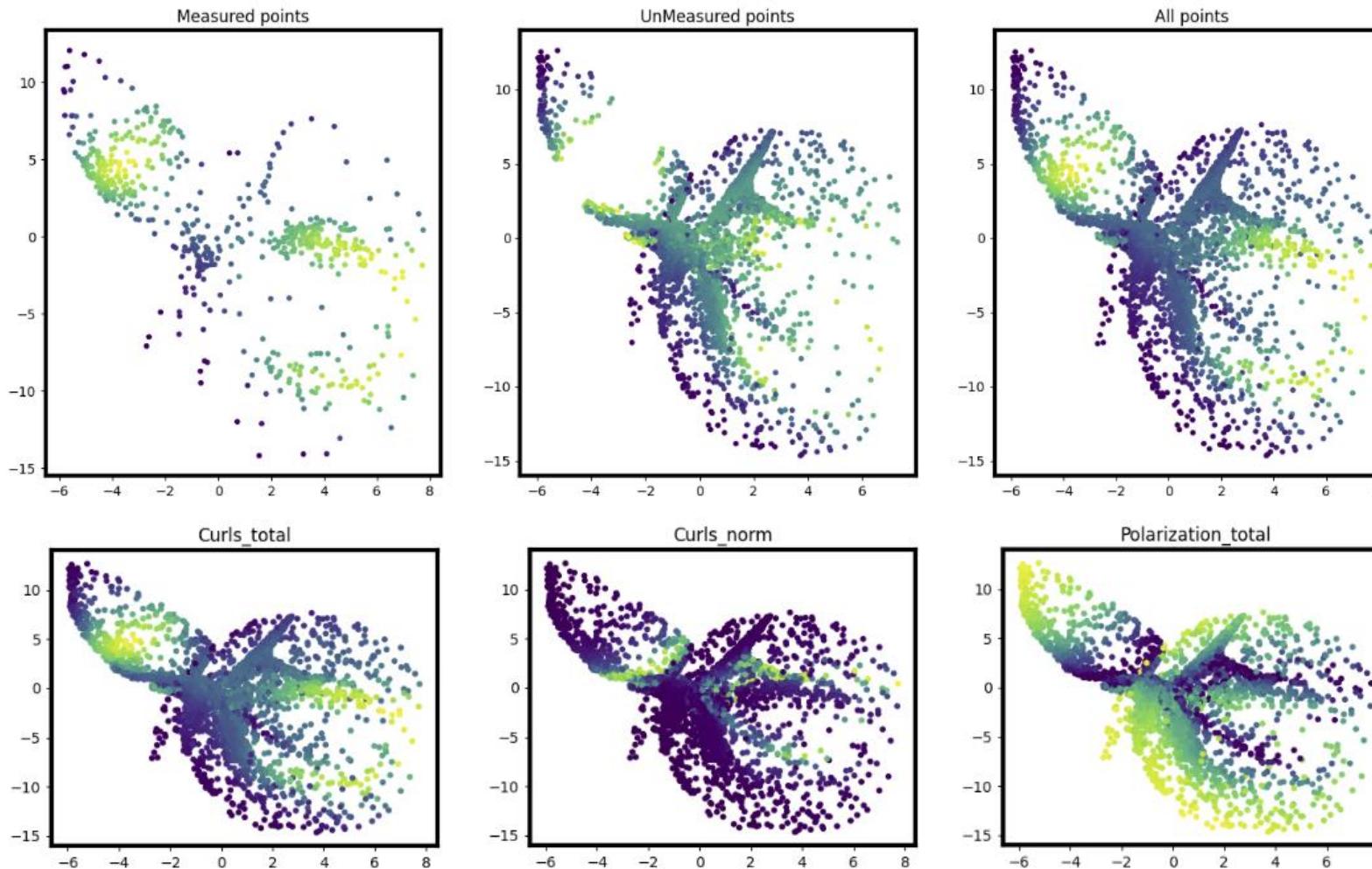


# DKL on FerroSIM: Static

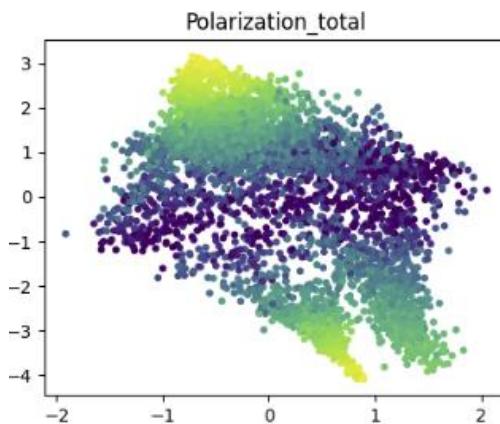
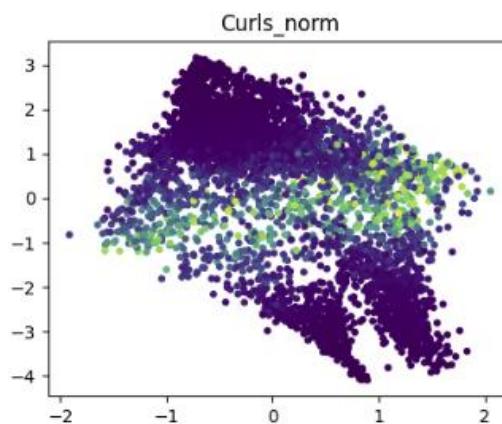
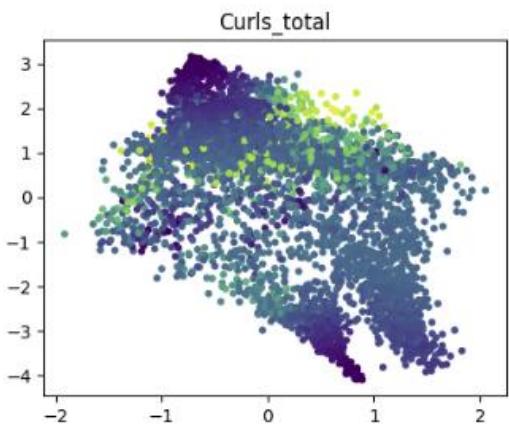
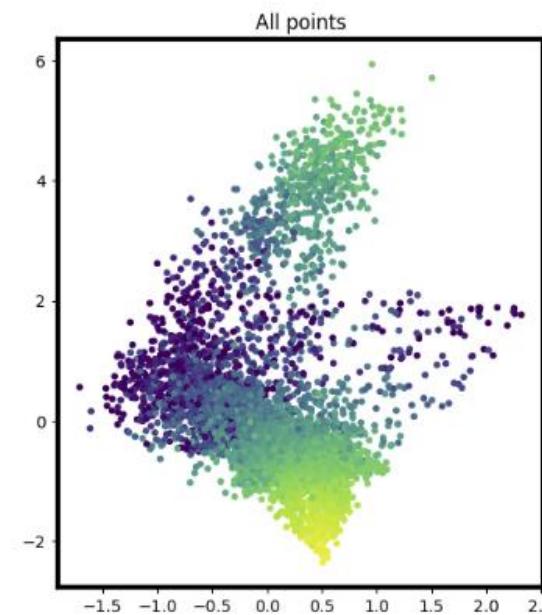
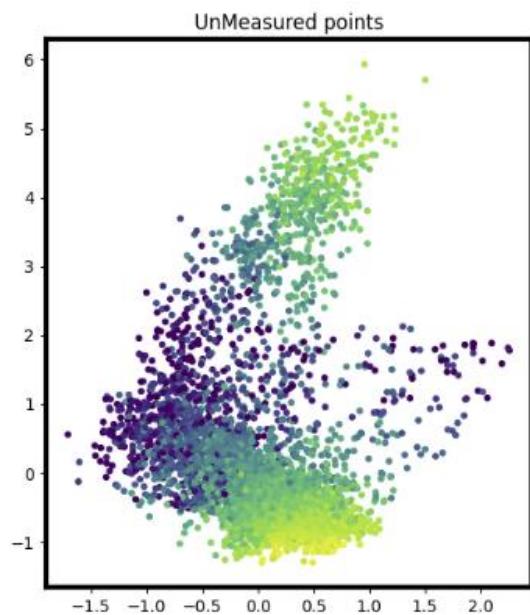
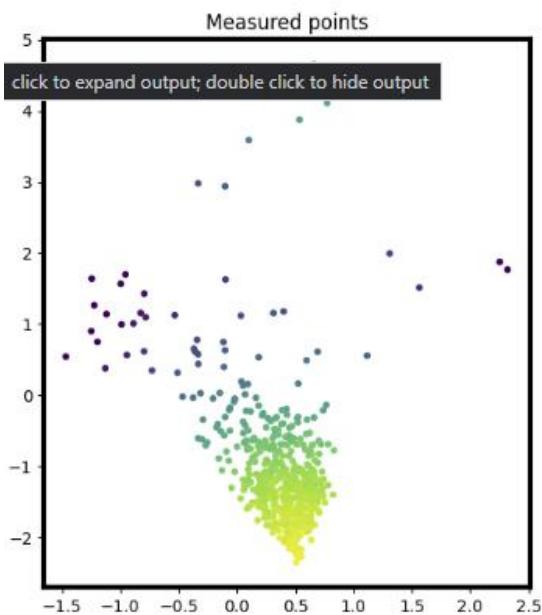
Target function:  
**Polarization**



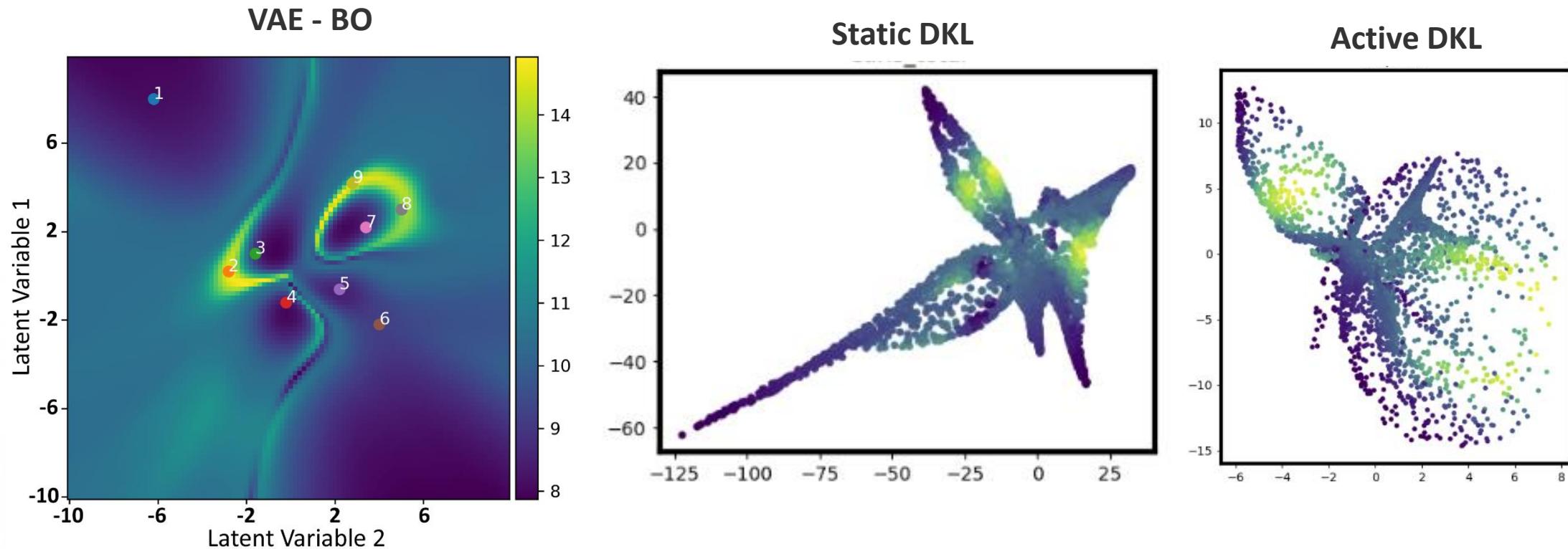
# DKL BO: Active Learning of Curl



# 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!