

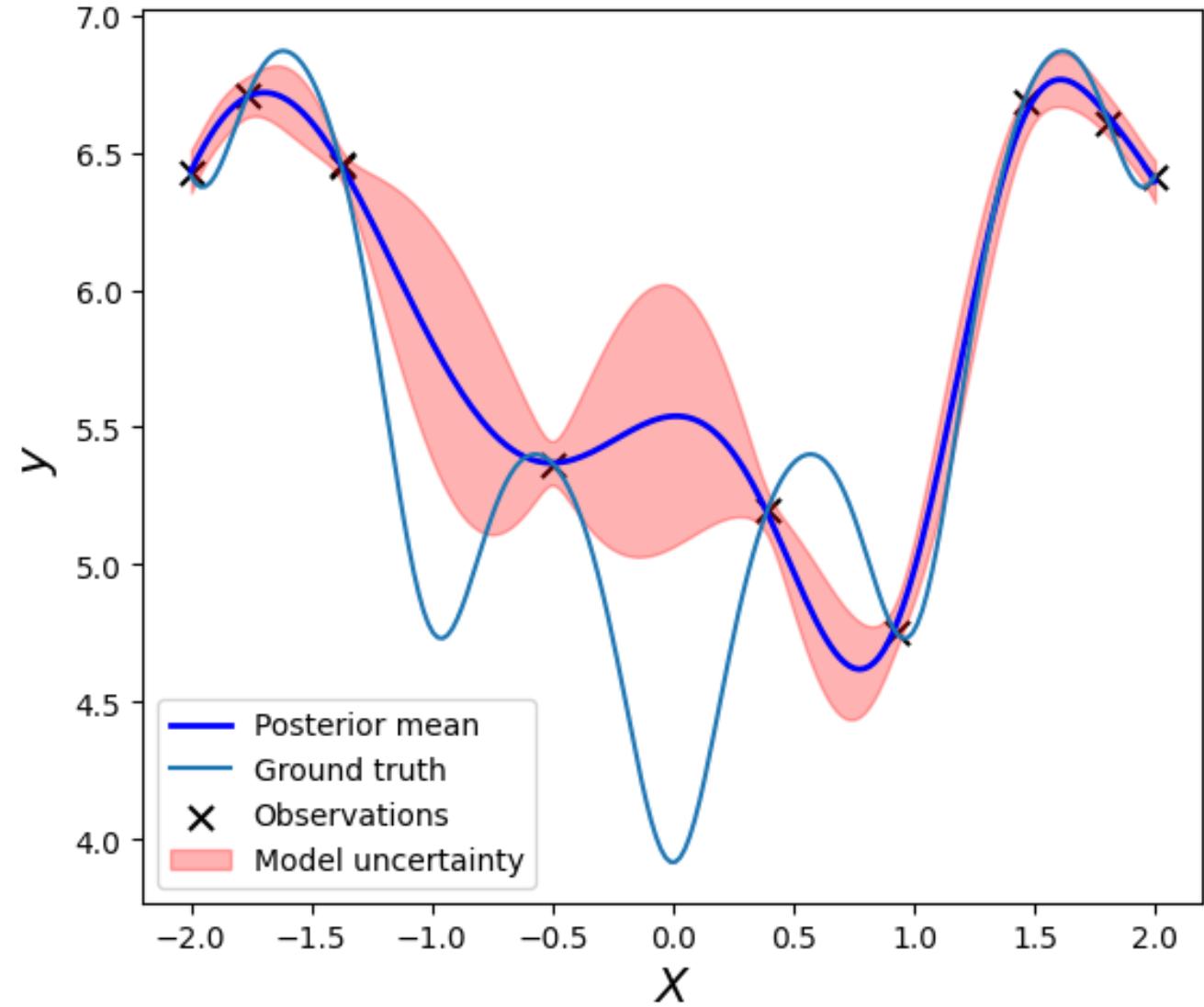
Bayesian Inference, Structured Gaussian Processes, and Hypothesis Learning

Sergei V. Kalinin

What have we learned from lecture 14

- Gaussian Process
- Kernel and kernel parameters
- Kernel Priors
- Noise Priors
- Posteriors

- Bayesian Optimization
- Bayesian Optimization based on Gaussian Process
- Acquisition Functions



What will we learn: Bayesian Inference Detour:

- Bayesian Inference
- Prior Distribution
- Posterior Distribution
- Least Square Fit
- WAIC

Colab Bayesian Inference

Automated Experiment: ... as a scientist...

Bayesian optimization:

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)

GP Augmented with Structural model

Define a probabilistic model:

$$\mathbf{y} \sim MVNormal(\mathbf{m}, \mathbf{K})$$

$$K_{ij} = \sigma^2 \exp(0.5(x_i - x_j)^2 / l^2)$$

$$\sigma \sim LogNormal(0, s_1)$$

$$l \sim LogNormal(0, s_2)$$

- We substitute a constant GP prior mean function \mathbf{m} with a structured probabilistic model of the expected system's behavior.
- This probabilistic model reflects our prior knowledge about the system, but it does not have to be precise.
- The model parameters are inferred together with the kernel parameters via the Hamiltonian Monte Carlo.
- The fully Bayesian treatment of the model allows additional control over the optimization via the selection of priors for the model parameters.

Prediction on new data X_* :

$$\mathbf{f}_*^i \sim MVNormal\left(\mu_{\boldsymbol{\theta}^i}^{\text{post}}, \Sigma_{\boldsymbol{\theta}^i}^{\text{post}}\right)$$

replaced with

$$\mu_{\boldsymbol{\theta}^i}^{\text{post}} = \mathbf{m}(X_*) + \mathbf{K}(X_*, X | \boldsymbol{\theta}^i) \mathbf{K}(X, X | \boldsymbol{\theta}^i)^{-1} (\mathbf{y} - \mathbf{m}(X)) \rightarrow \mu_{\Omega^i}^{\text{post}} = \mathbf{m}(X_* | \phi^i) + \mathbf{K}(X_*, X | \boldsymbol{\theta}^i) \mathbf{K}(X, X | \boldsymbol{\theta}^i)^{-1} (\mathbf{y} - \mathbf{m}(X | \phi^i))$$

$$\Sigma_{\boldsymbol{\theta}^i}^{\text{post}} = \mathbf{K}(X_*, X_* | \boldsymbol{\theta}^i) - \mathbf{K}(X_*, X | \boldsymbol{\theta}^i) \mathbf{K}(X, X | \boldsymbol{\theta}^i)^{-1} \mathbf{K}(X, X_* | \boldsymbol{\theta}^i)$$

$\Omega^i = \{\phi^i, \boldsymbol{\theta}^i\}$ is a single HMC posterior sample with the kernel and prob model parameters

GP Augmented with Structural model

E

Probabilistic model

$$m = y_0 - \sum_{n=1}^N L_n \quad (N=2)$$

$$y_0 \sim \text{Uniform}(-10, 10)$$

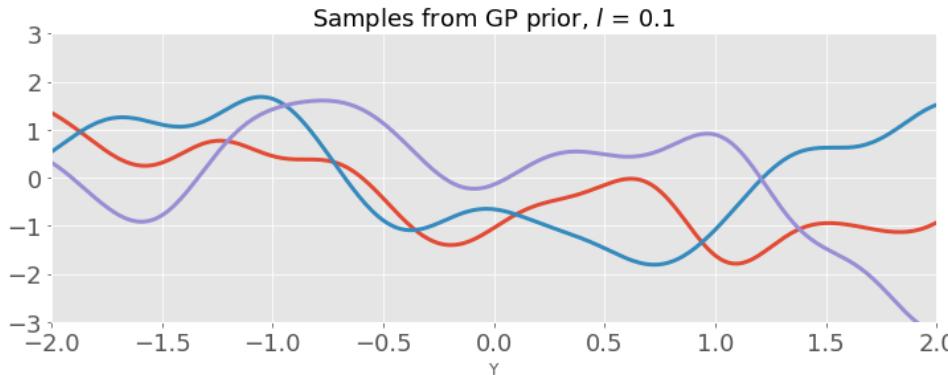
$$L_n \sim \frac{A_n}{\sqrt{(x-x_n^0)^2+w_n^2}}$$

$$A_n \sim \text{LogNormal}(0, 1)$$

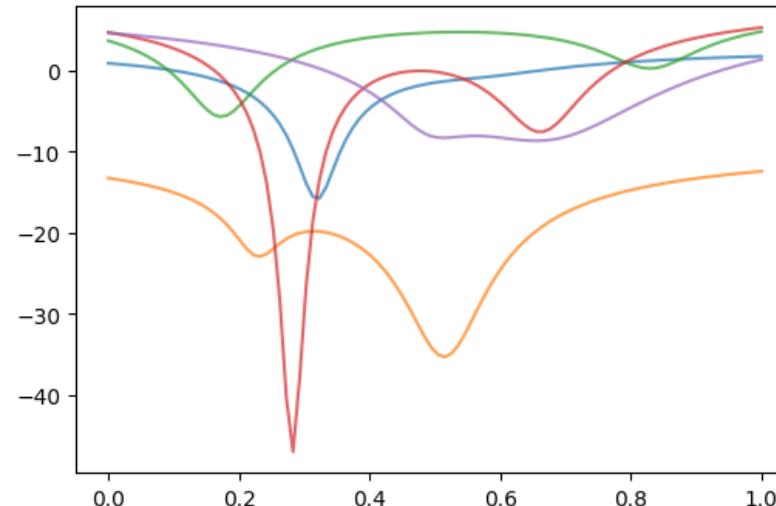
$$w_n \sim \text{HalfNormal}(.1)$$

$$x_n^0 \sim \text{Uniform}(0, 1)$$

Prior predictive distribution: GP

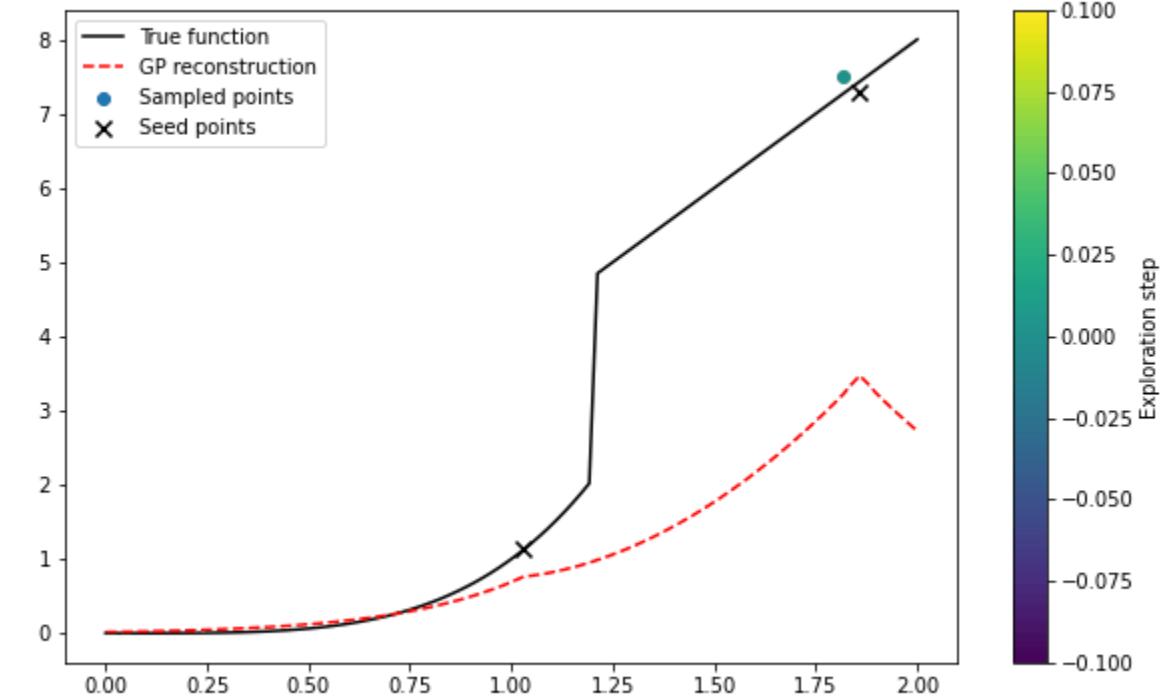
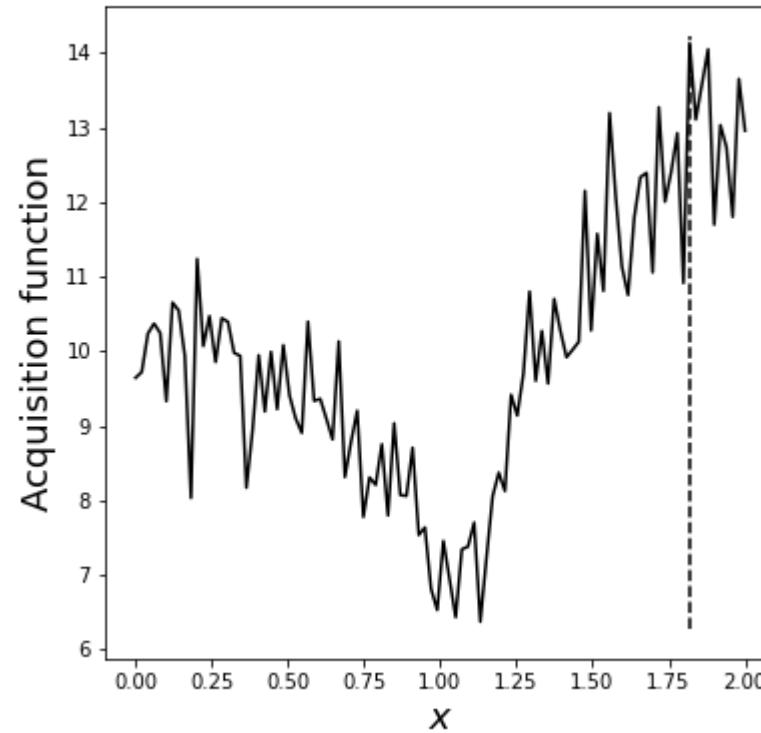


Prior predictive distribution: sGP

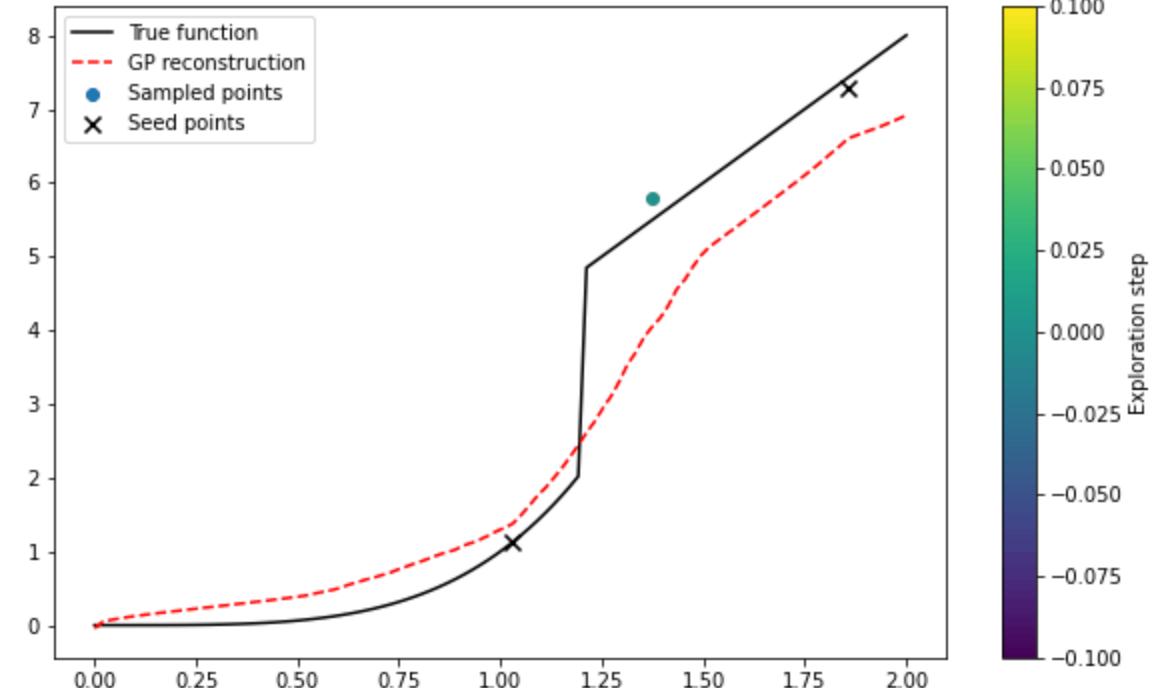
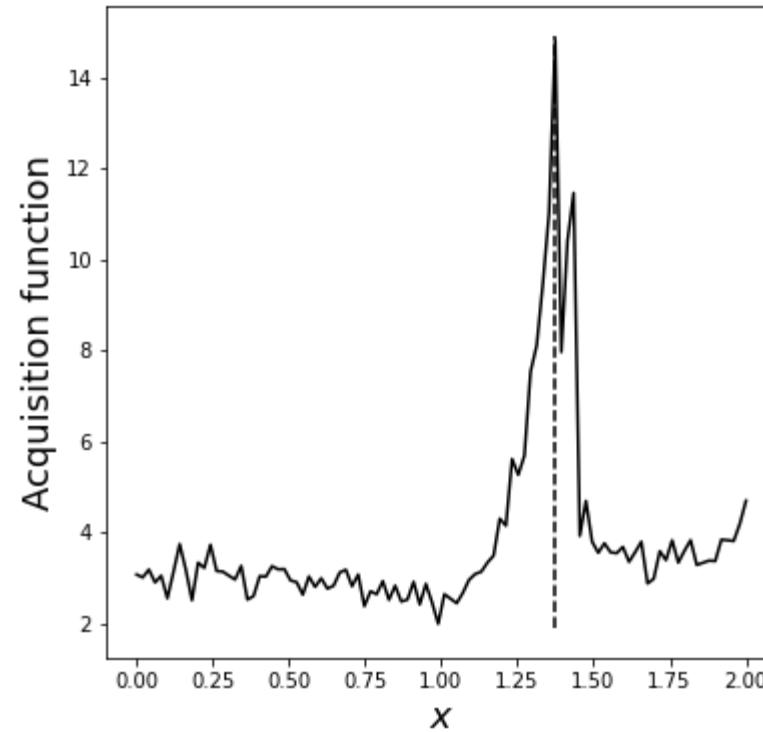


This model simply tells us that there are two minima in our data but does not assume to have any prior knowledge about their relative depth, width, or distance

Simple GP search



Structured GP search

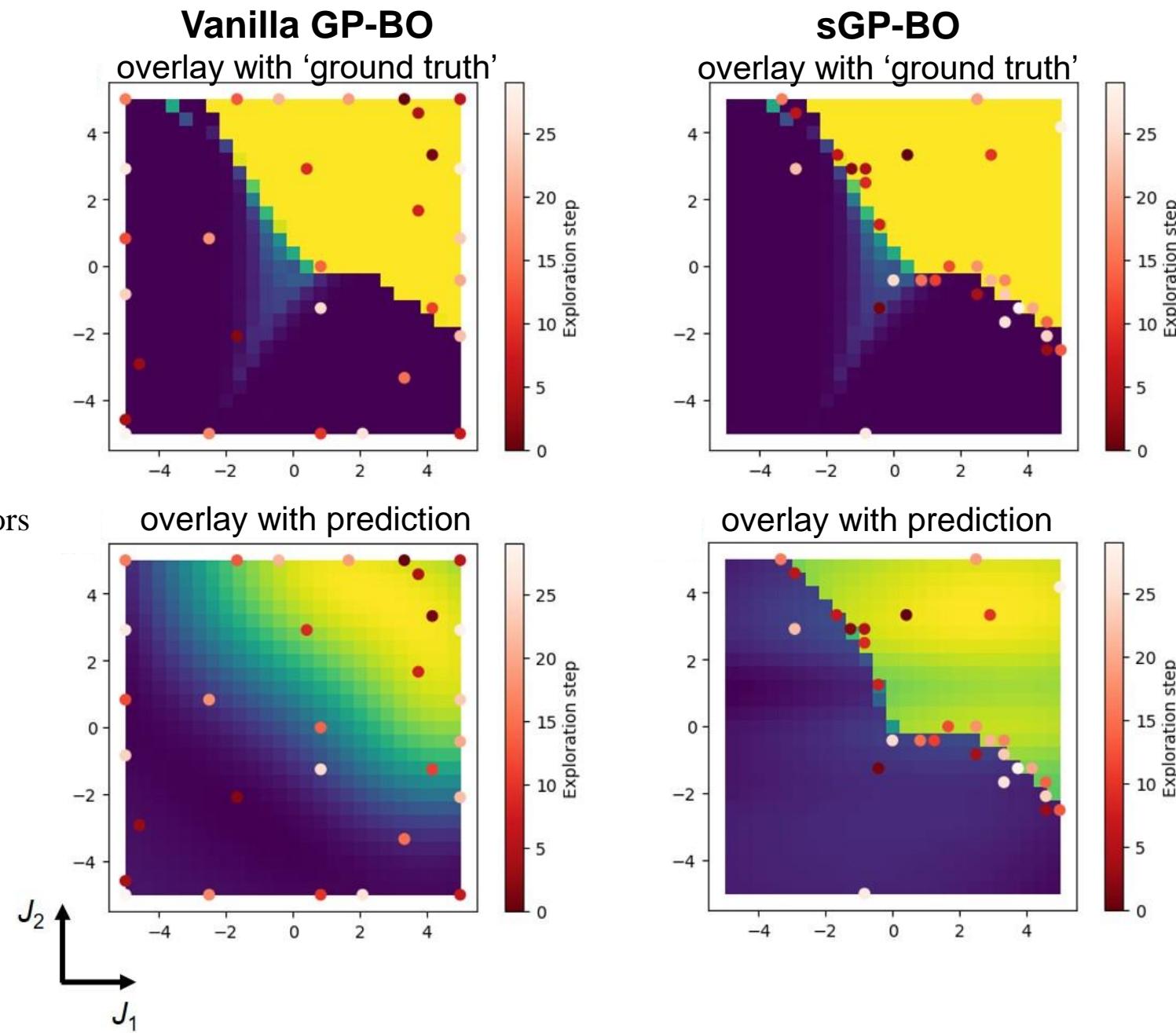


Application to Ising model

Probabilistic model

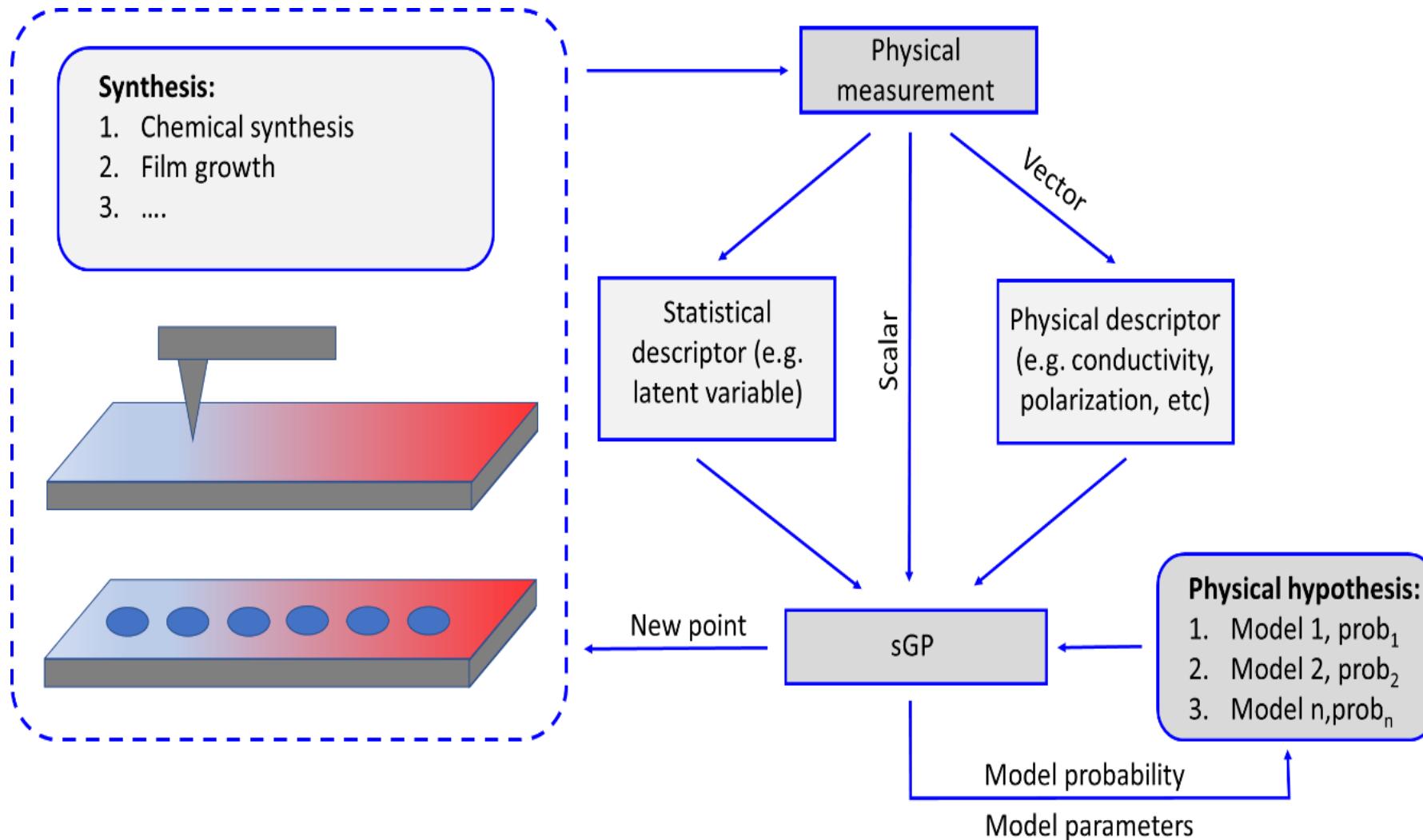
$$A/\tanh\left(\frac{f(J_1)+f(J_2)}{w}\right)$$

where $f(J)$ is a third-degree polynomial with normal priors on its parameters

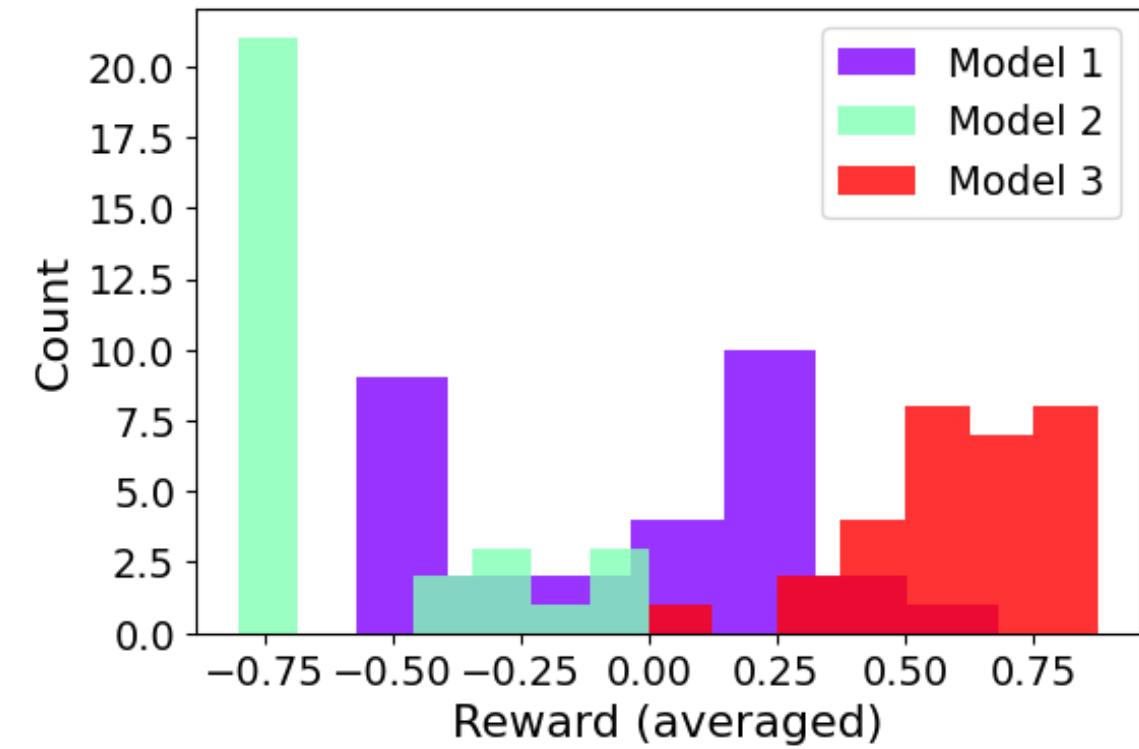
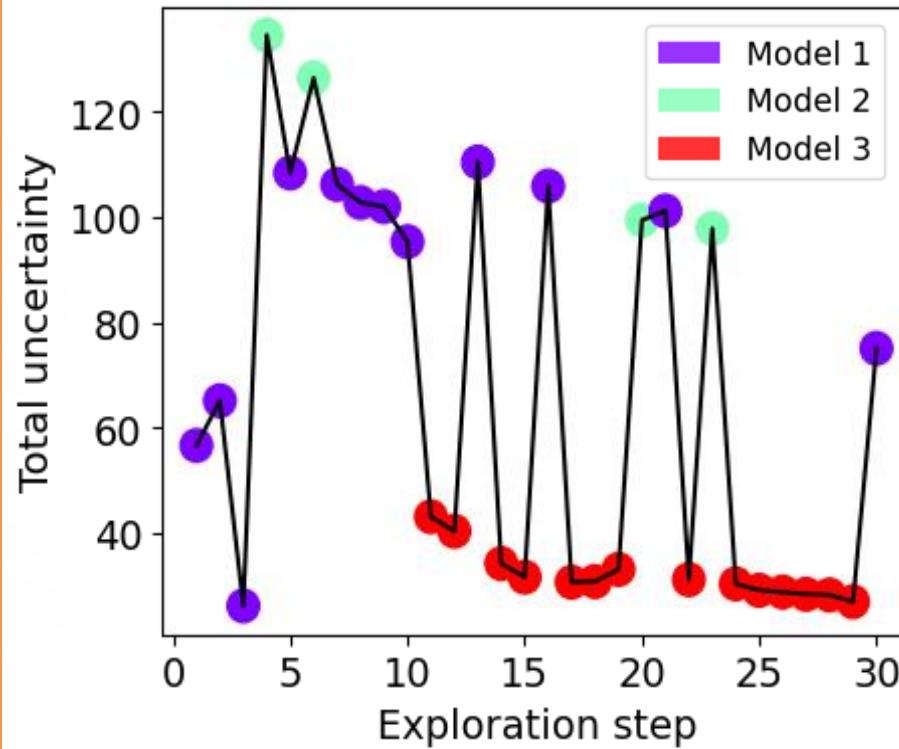


Colab 2

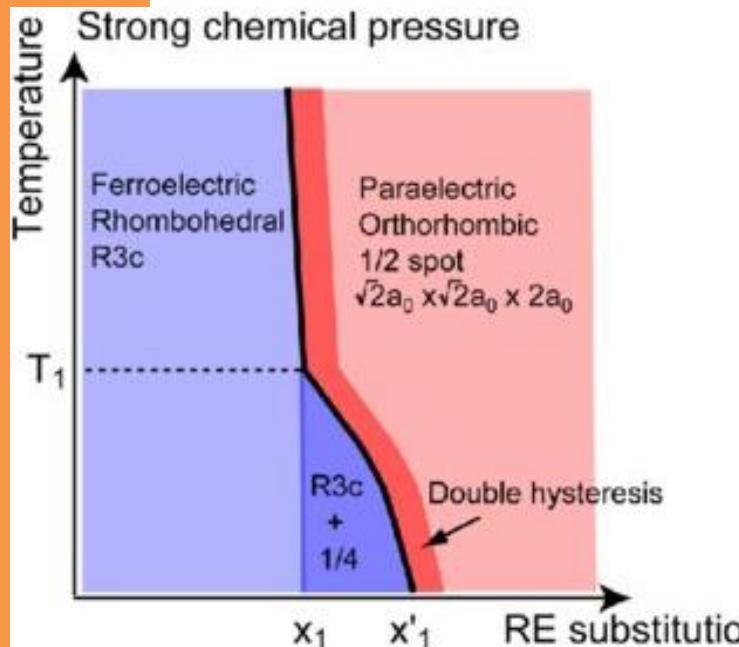
Hypothesis active learning: hypoAL



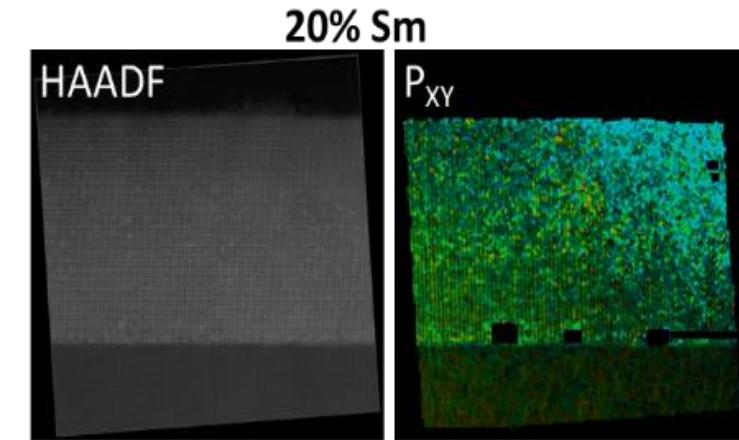
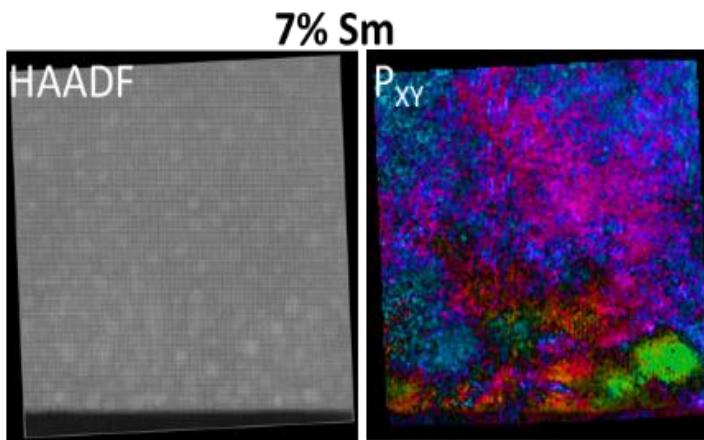
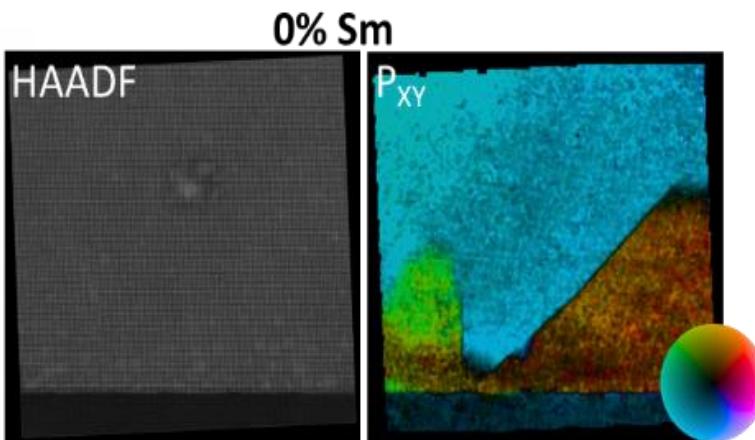
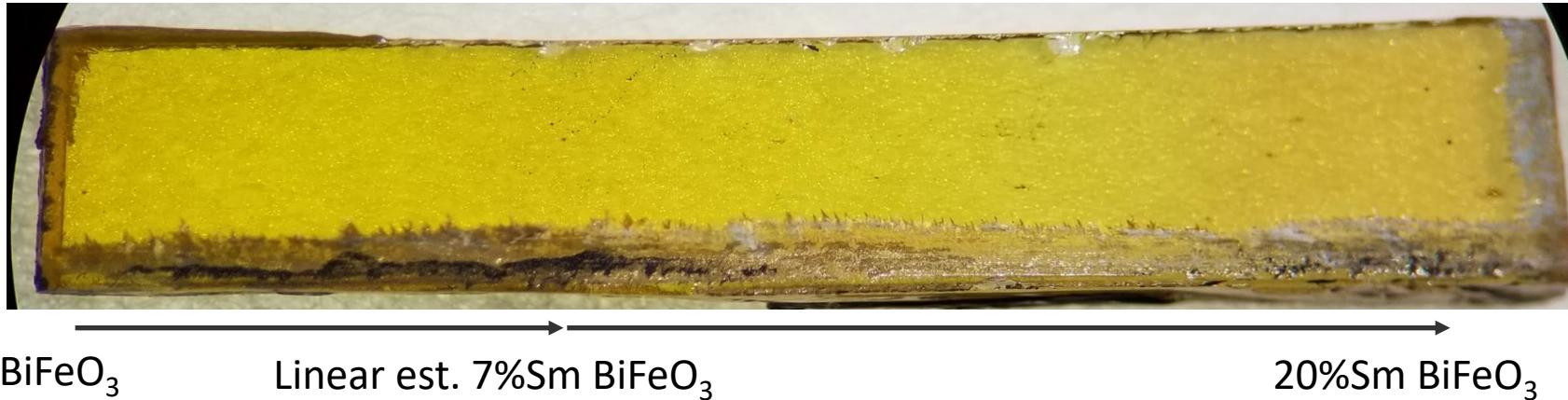
Next step: active model selection



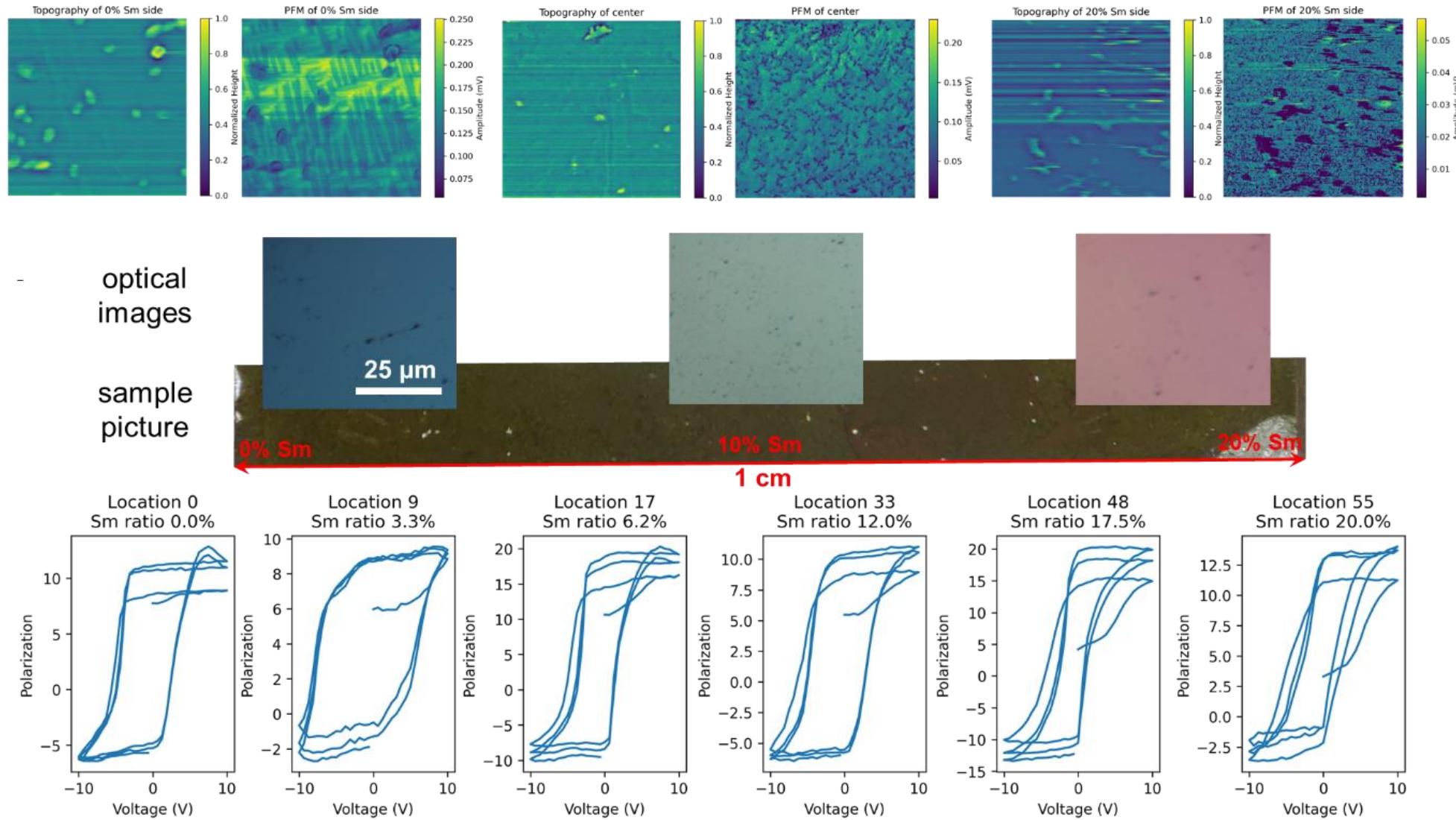
Combinatorial Synthesis



Sample by I. Takeuchi, UMD
Phase diagram by N. Valanoor et al.



Combinatorial libraries:



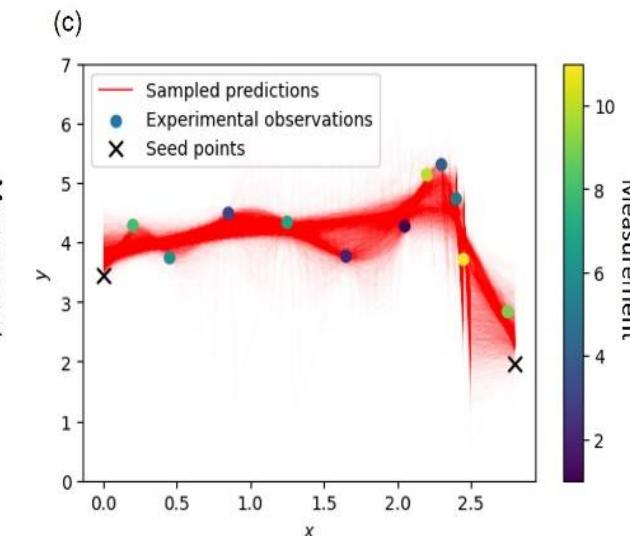
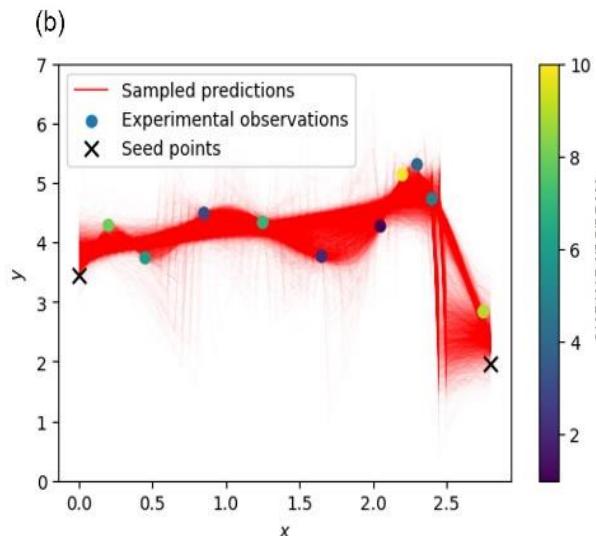
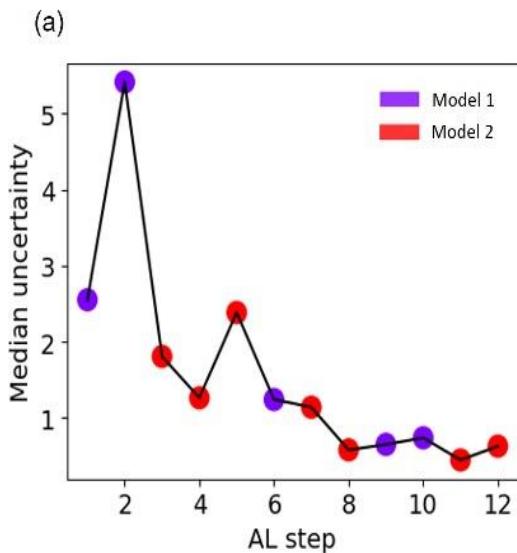
Hypothesis selection for ferroelectric response

Model 1 (second order phase transition):

$$S = \begin{cases} S_0 \left(1 - \frac{x}{x_0}\right)^2 + C, & x \leq x_c, \\ C, & x > x_c \end{cases}$$

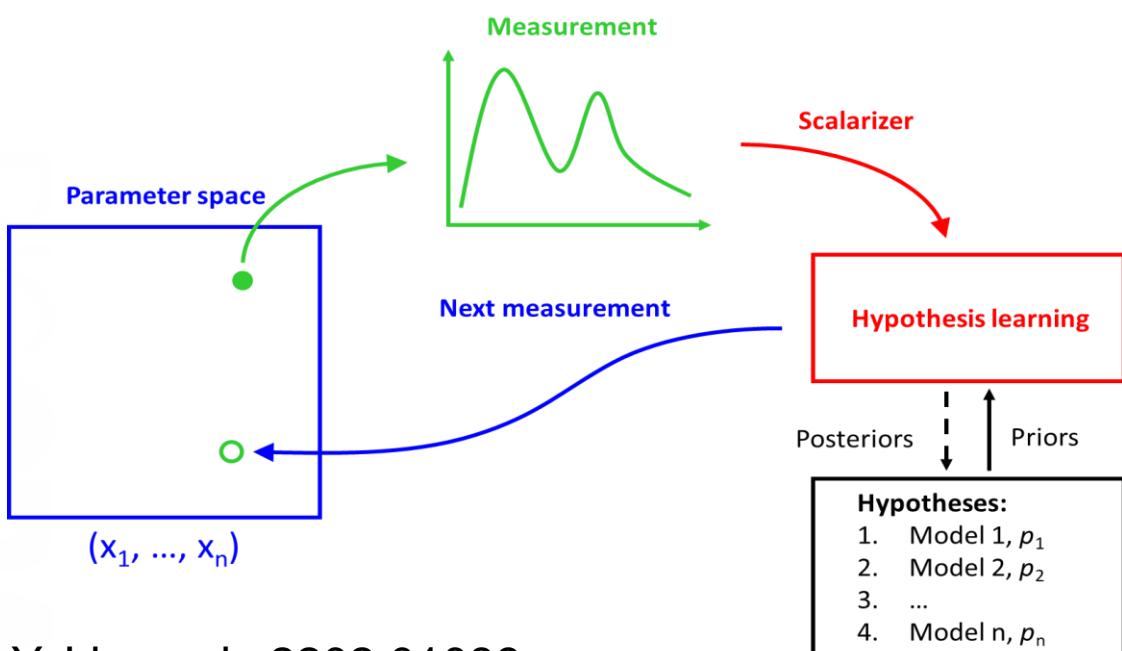
Model 2 (first order phase transition):

$$S = \begin{cases} S_0 \left(1 - \frac{x}{x_0}\right)^{\frac{5}{4}} + C_0, & x \leq x_c, \\ C_1, & x > x_c \end{cases}$$



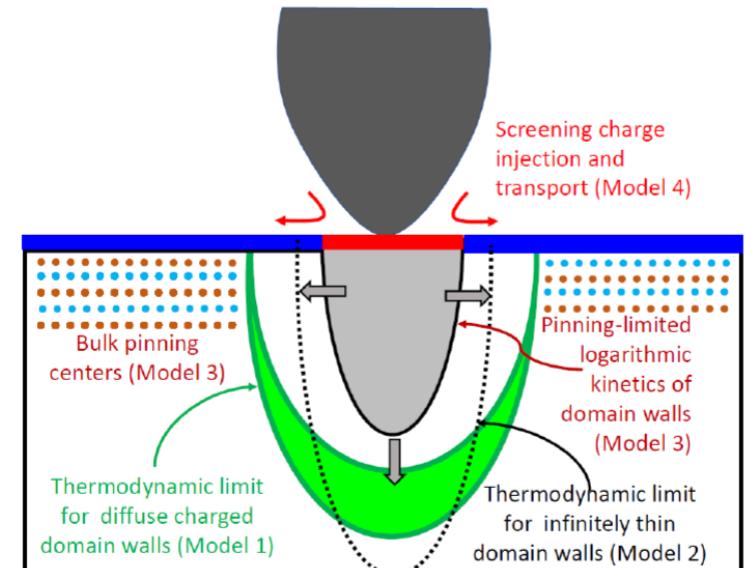
Hypothesis learning for domain growth laws

- Can ML algorithm think like a scientist?
- Yes – automated experiment can pursue hypothesis-driven science!

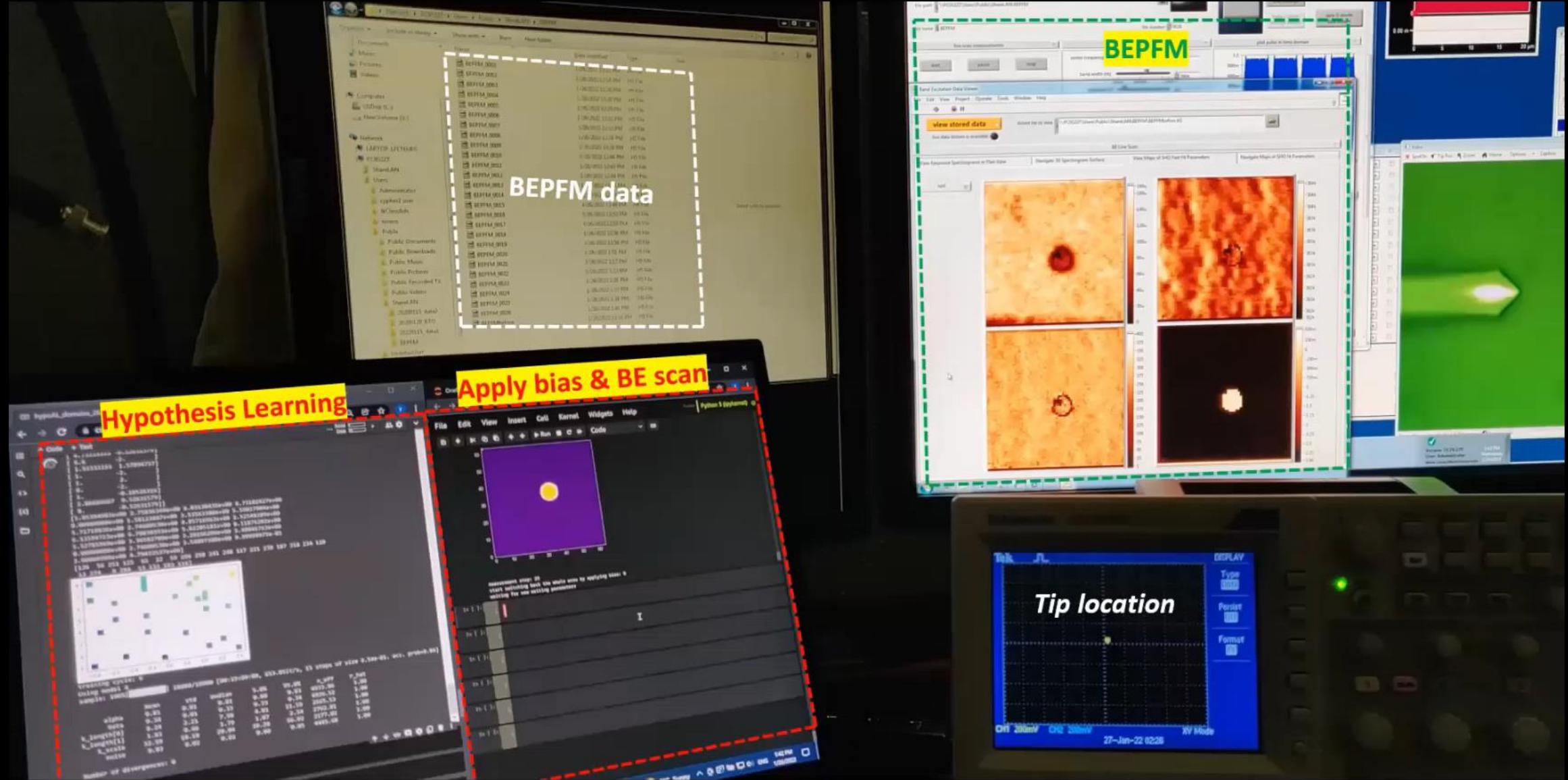


Y. Liu, arxiv 2202.01089

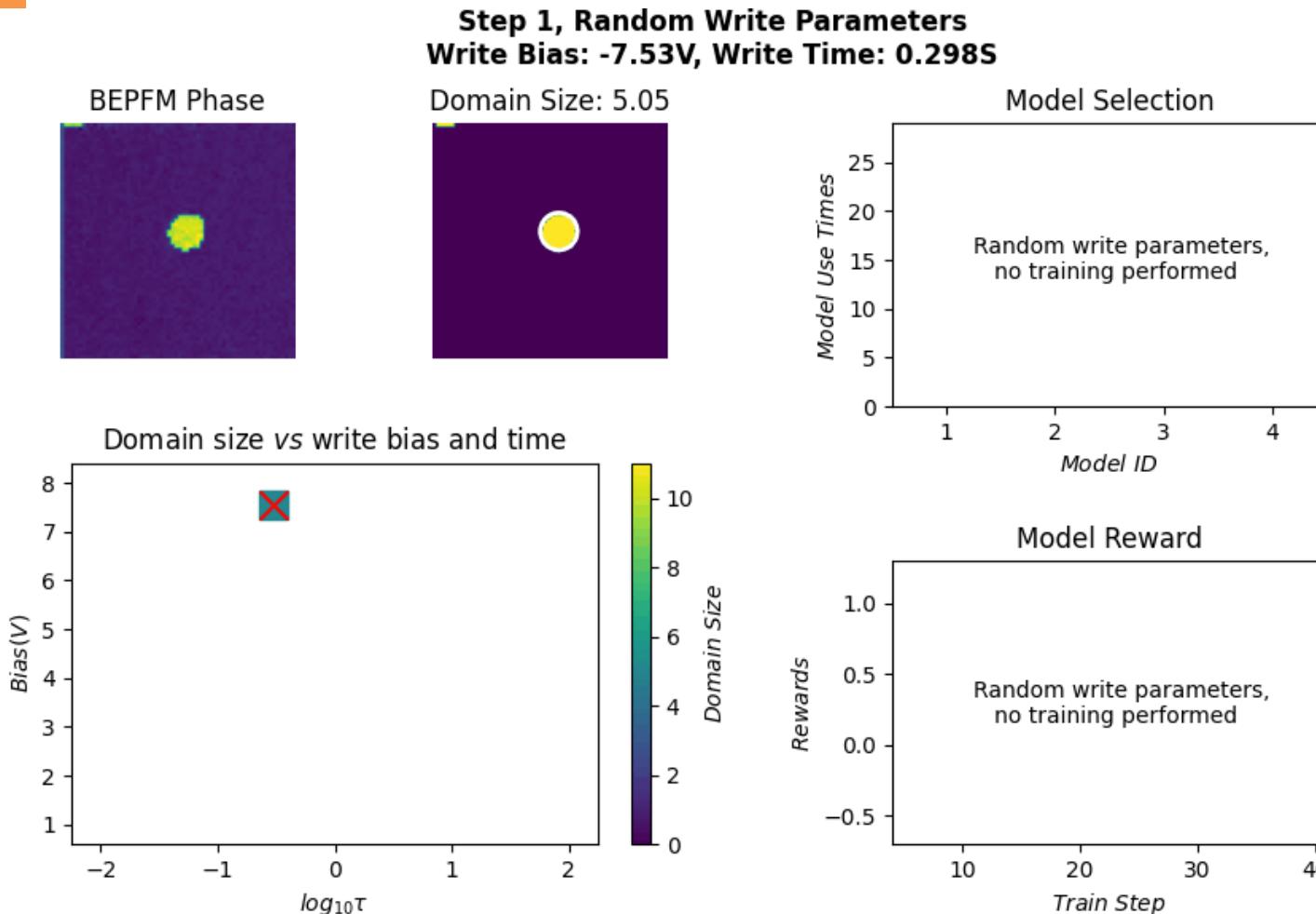
Y. Liu, arxiv 2112.06649



Model Equation		
Thermodynamic 1	Model I	$r(V) = r_{cr} + r_0 \sqrt{\left(\frac{V}{V_c}\right)^{2/3} - 1}$
Thermodynamic 2	Model II	$r(V) = r_{cr} + r_0 \sqrt[3]{\left(\frac{V}{V_c}\right)^2 - 1}$
Wall pinning	Model III	$r(V, t) = V^\alpha \log \tau$
Charge injection	Model IV	$r(V, t) = V^\alpha \tau^\beta$

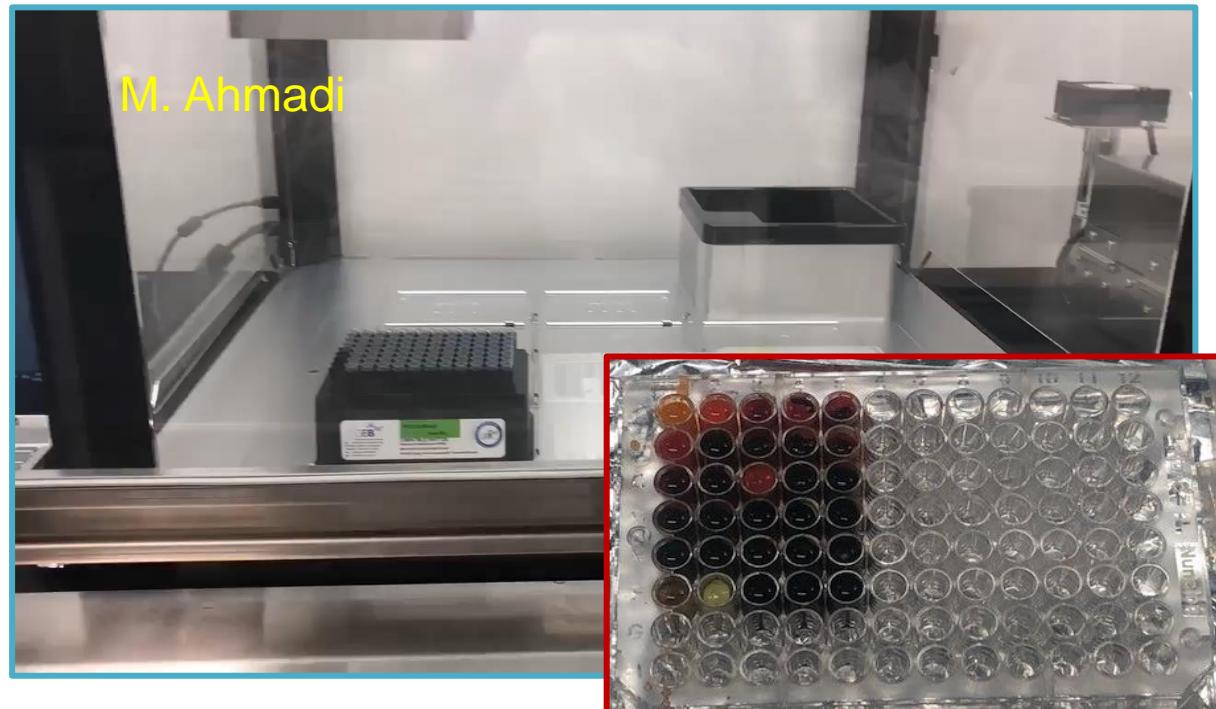
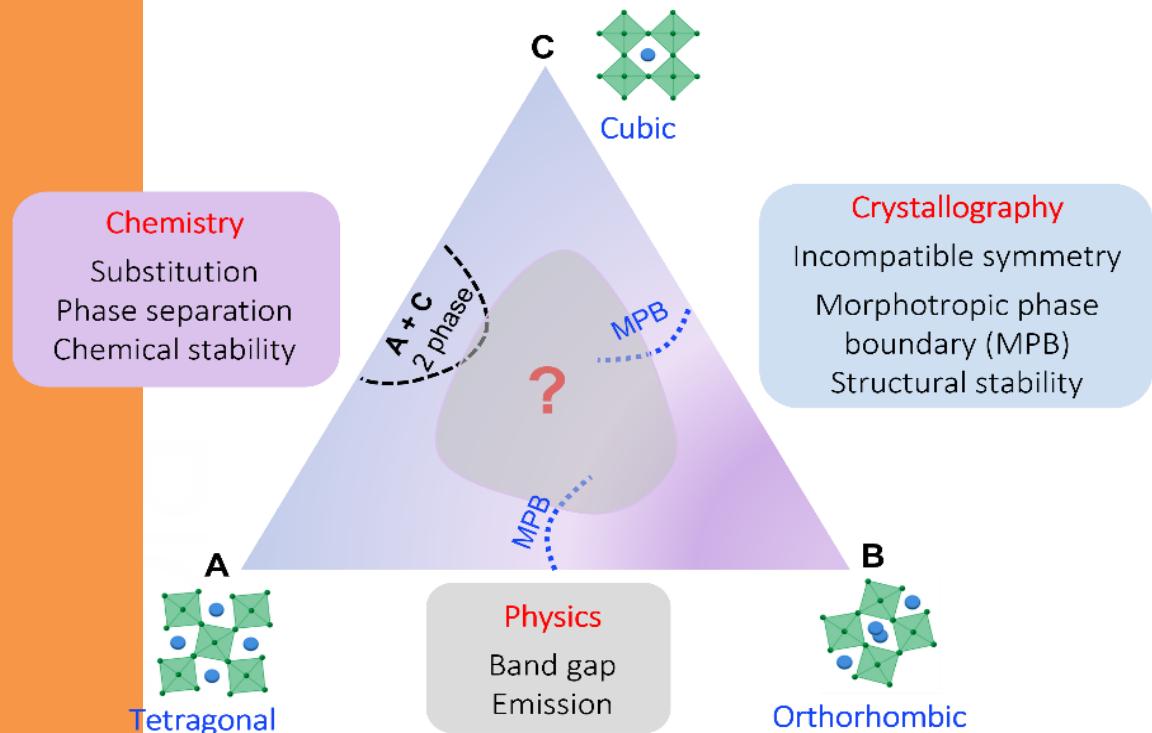


Hypothesis learning in action

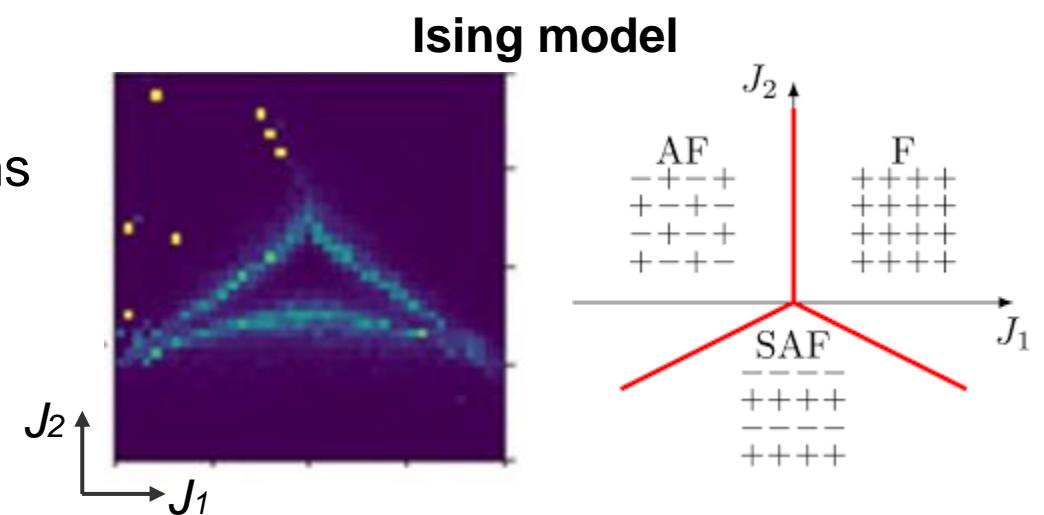


- ML algorithm has 4 competing hypothesis on domain switching mechanisms
- These hypothesis represent full set of possibilities for this system
- The microscope chooses experimental parameters in such a way as to establish which hypothesis is correct fastest
- Important: the same approach can be implemented in synthesis and electrical characterization
- Machine learning meets hypothesis-driven scientific discovery!

Why synthesis (or theory)?



- Automated synthesis in its simplest form requires some way to navigate phase diagrams
- In more complex form, processing space.
- Ideally, incorporate physical knowledge
- Similar problem - theory



The World is Bayesian

Hypothesis driven science:
What we want to learn

Forward model: Theory

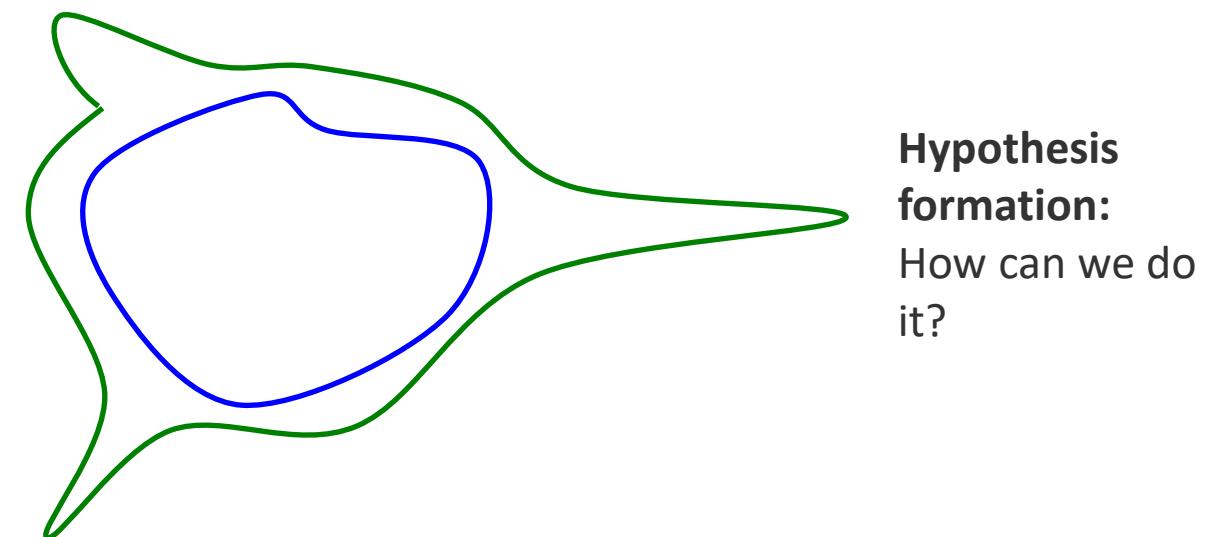
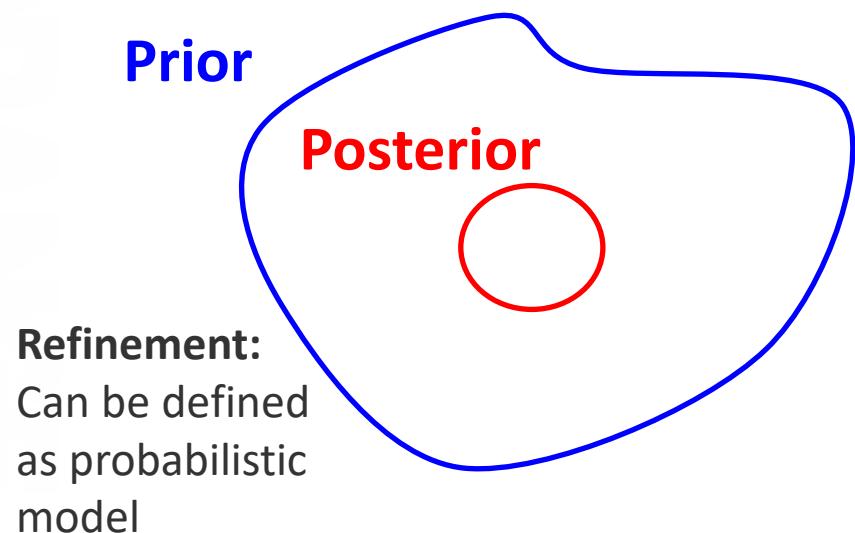
Domain expertise:

$$P(\text{Theory}|\text{Data}) = \frac{P(\text{Data}|\text{Theory})P(\text{Theory})}{P(\text{Data})}$$

Experimentalists know the priors. Albeit they do not know that they know it, or how to convert them to algorithmic form

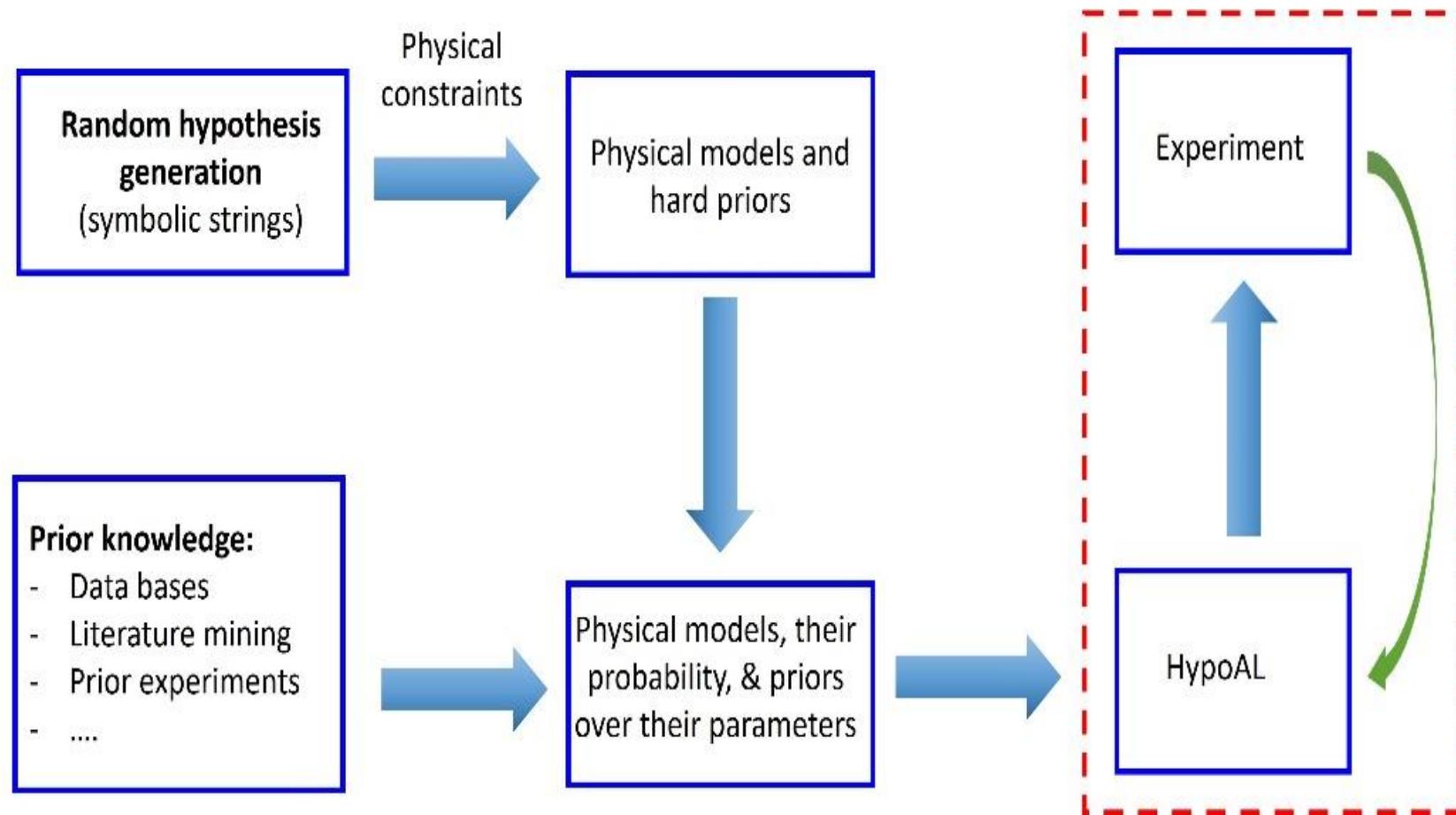
High Performance Computing

However, how do we make guesses about the unknown?



Towards hypothesis learning

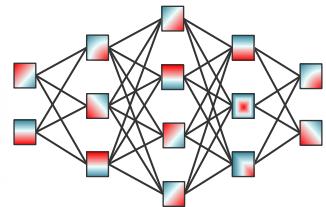
- Data already exists: Eureka, SISSO, SinDY, etc.
- What if we make hypothesis learning a part of active experiment?
- Need policies for hypothesis generation



Human in the Loop: Hypothesis injection

Prior knowledge

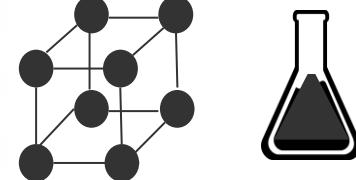
- Weights
- Inductive biases



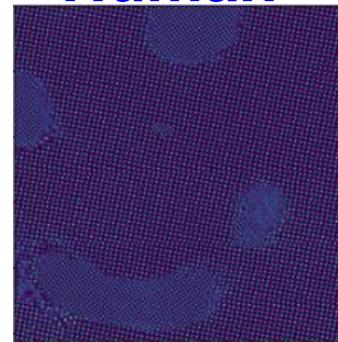
Reward Function



Material Parameters

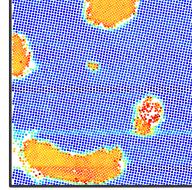


Human

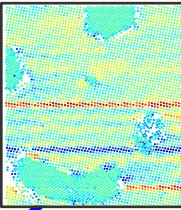
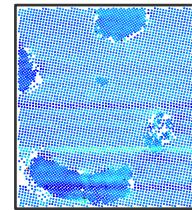


Experiment

Select analysis method



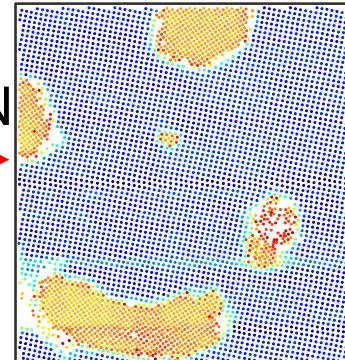
Select component



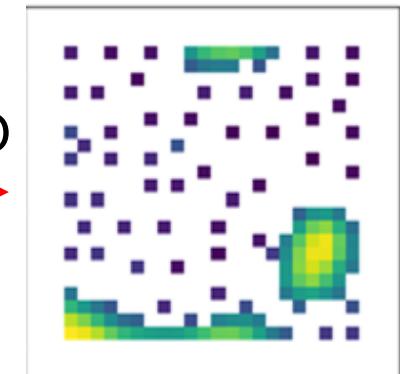
AI



DCNN



rVAE ● BO



Discovery pathway depends on the reward structure!