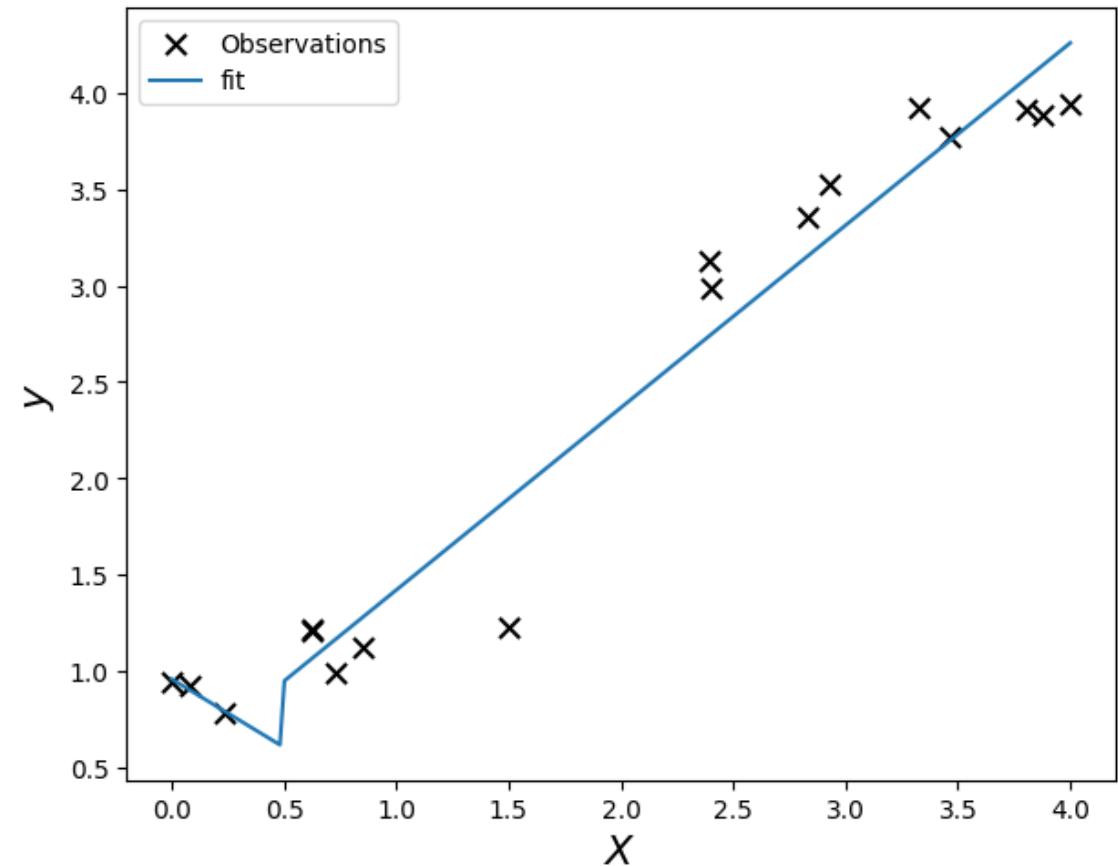
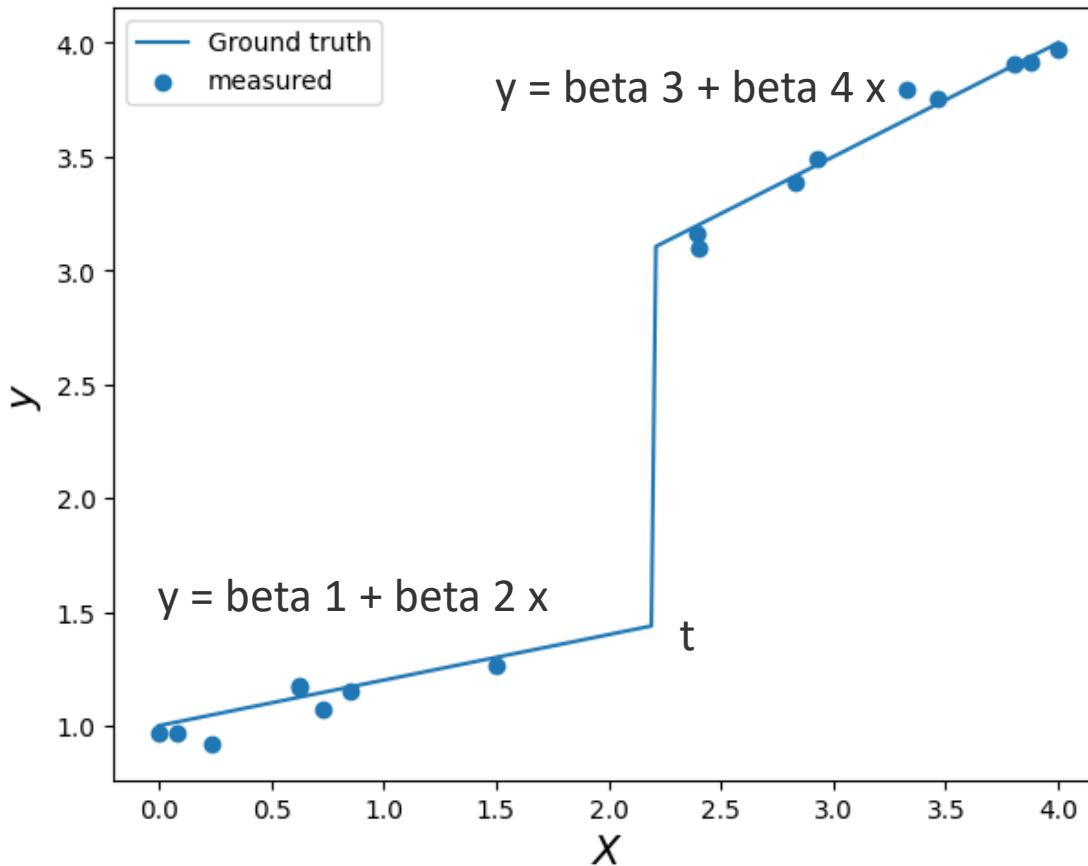


Lecture 21: Bayesian Inference and Gaussian Processes

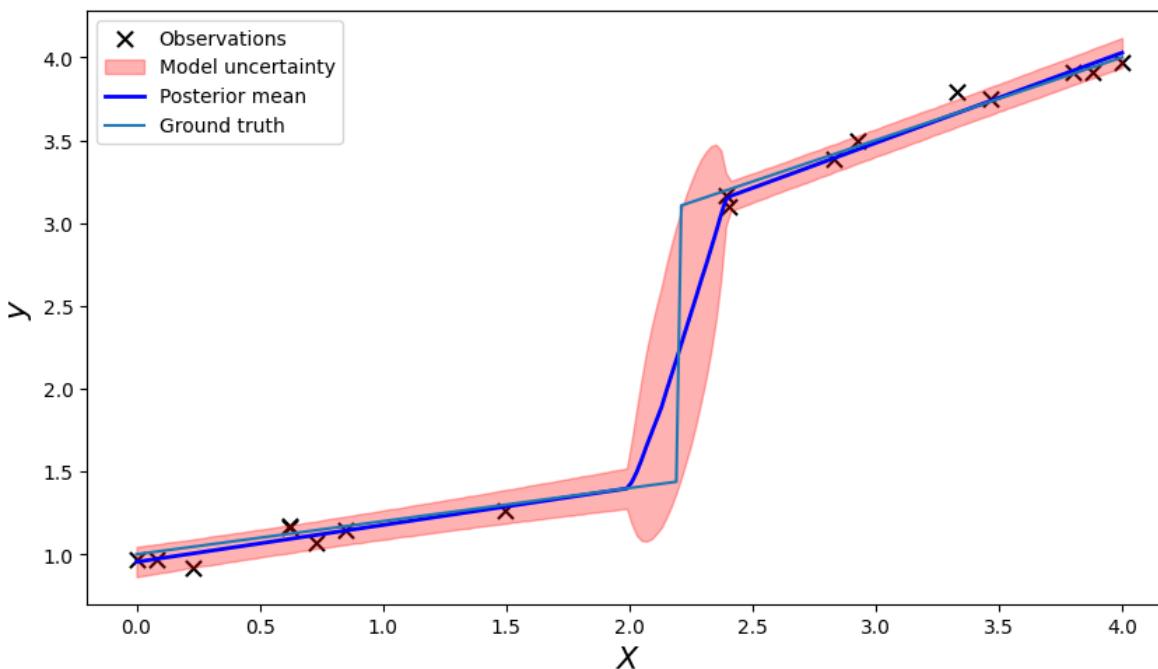
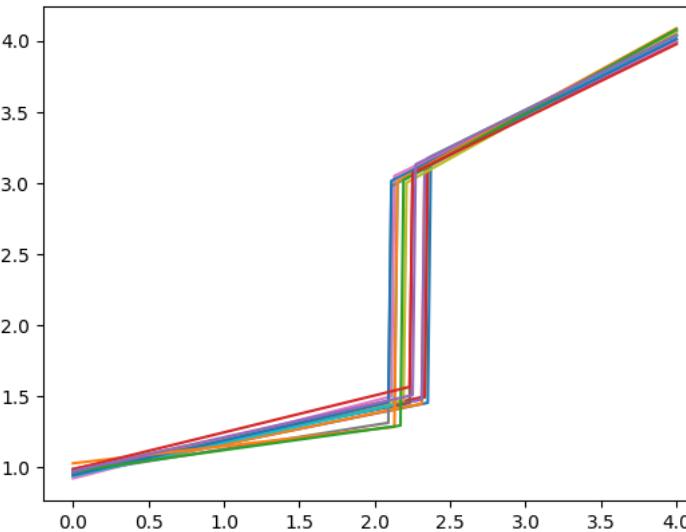
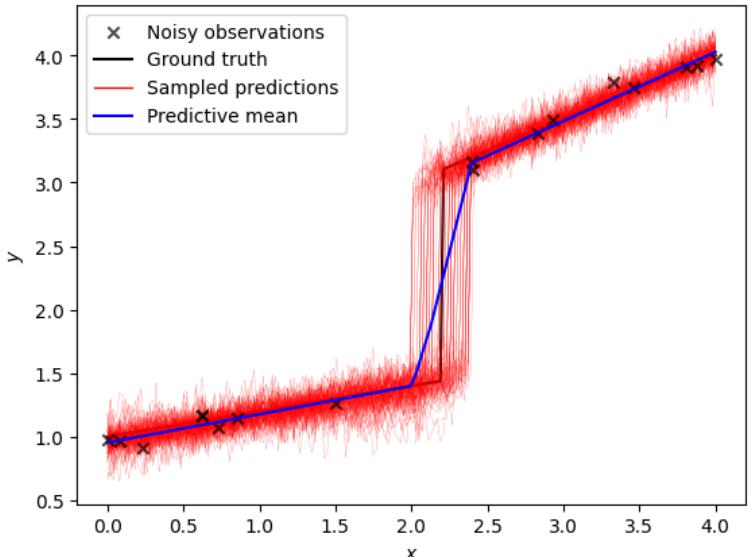
Instructor: Sergei V. Kalinin

Least Square Fits can be a problem



- No way to incorporate prior knowledge
- Convergence to metastable minima
- No feedback when gradient descent gets stuck

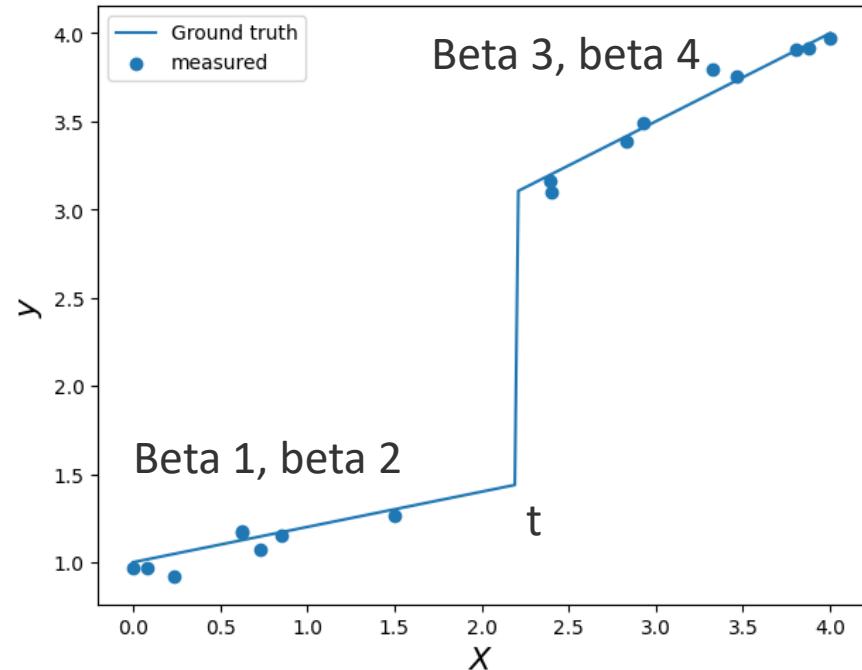
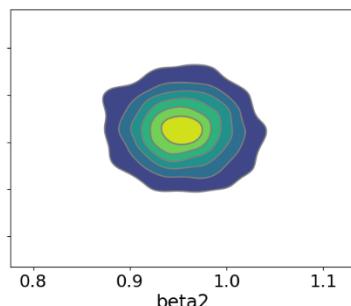
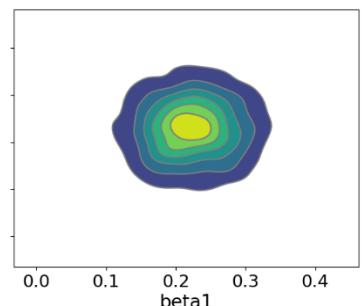
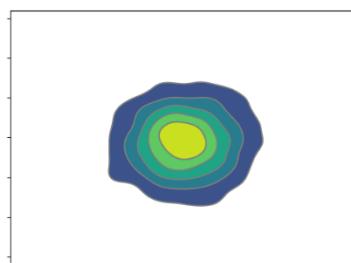
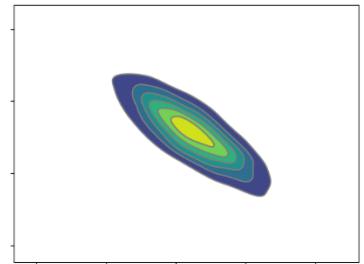
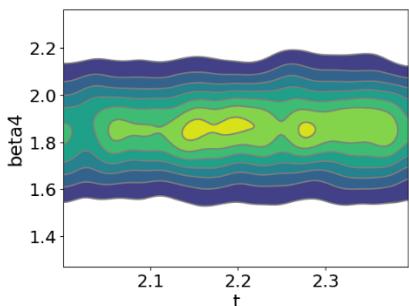
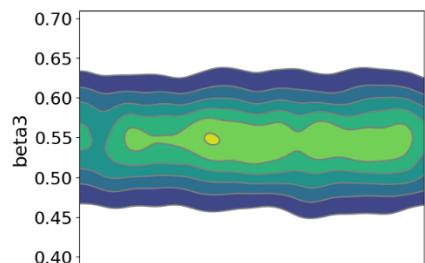
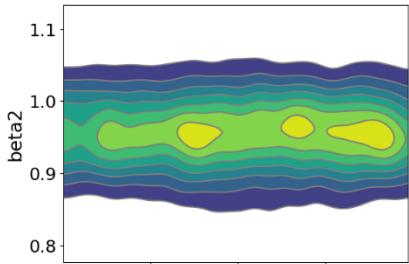
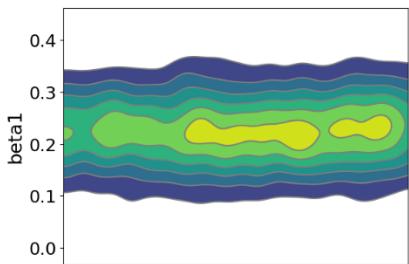
Bayesian Inference



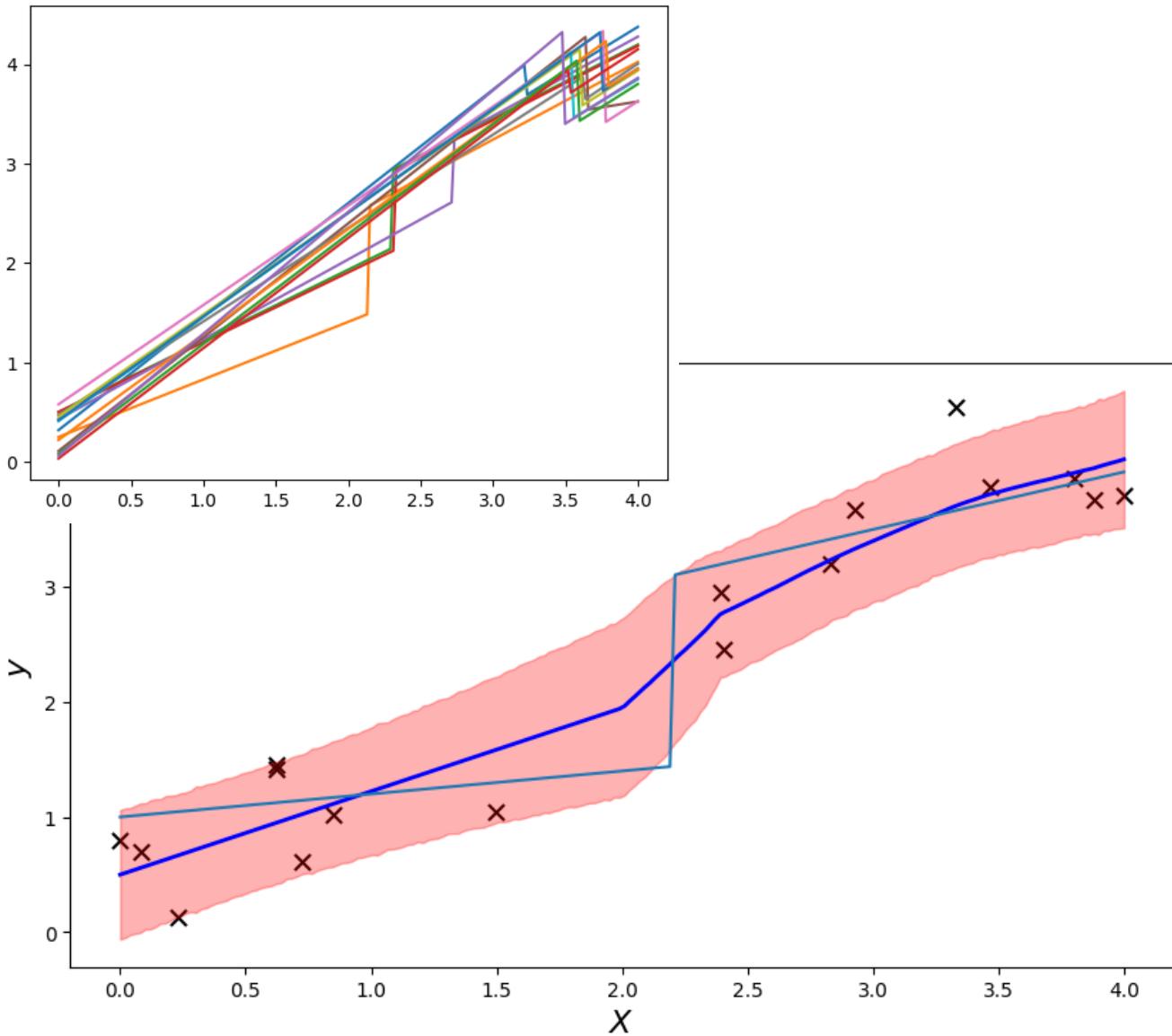
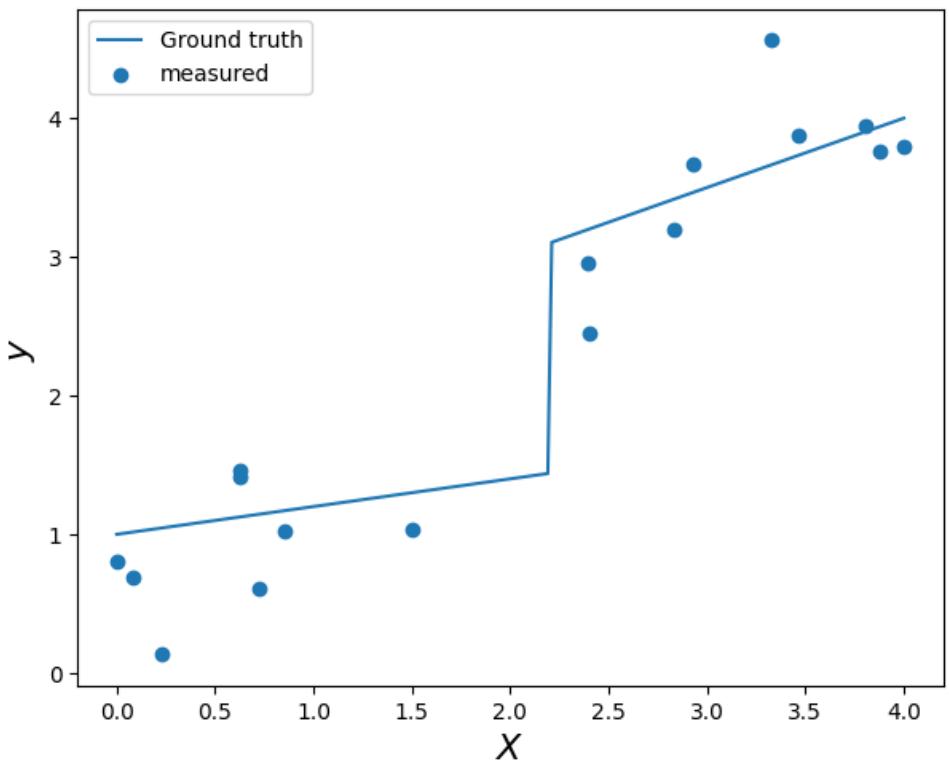
Bayesian inference:

- Priors on parameter values
- Sampling values or mean functions
- Predictive mean and uncertainty
- Posteriors on parameter values

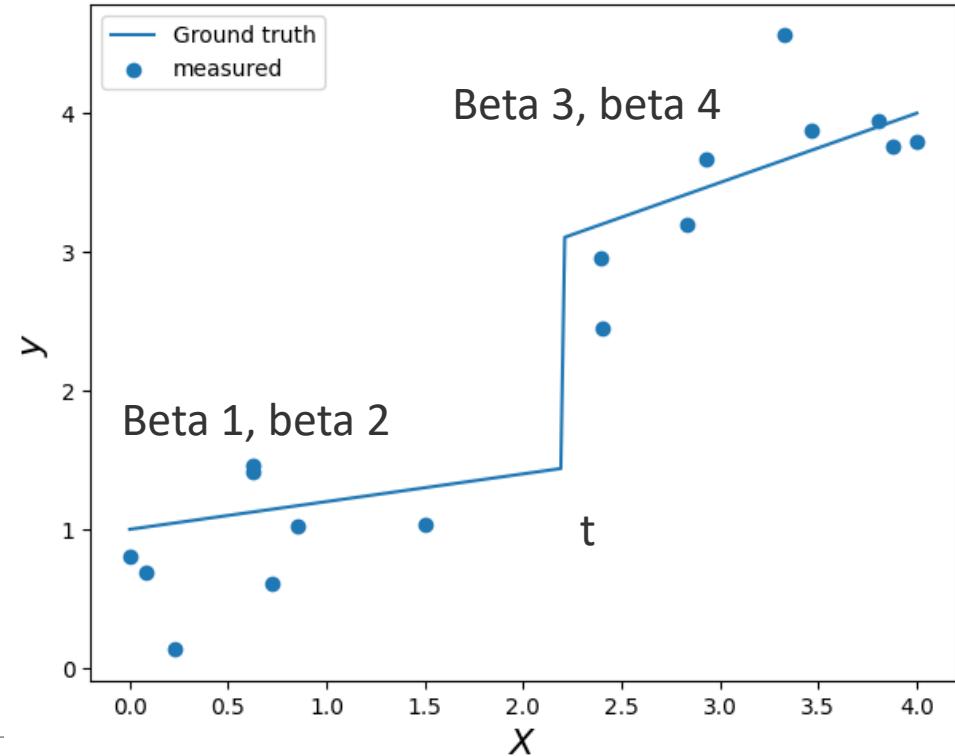
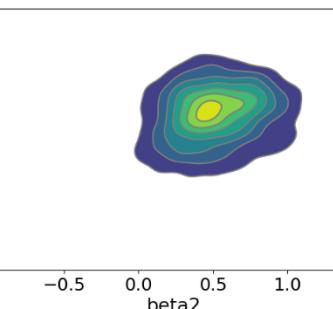
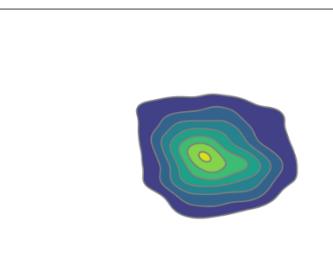
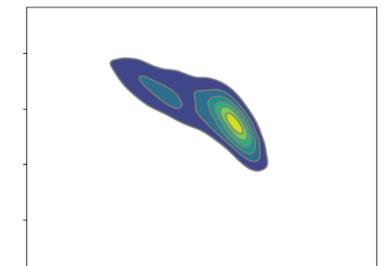
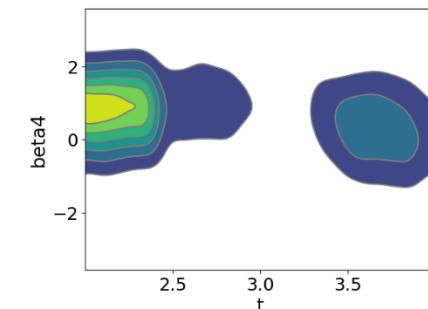
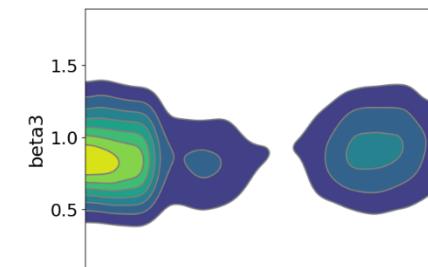
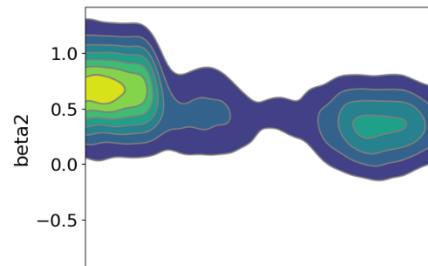
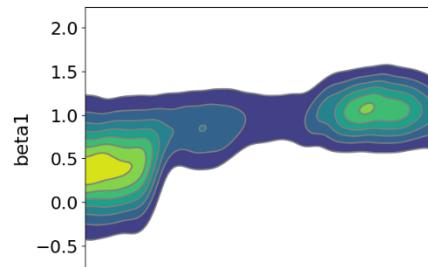
Bayesian Inference



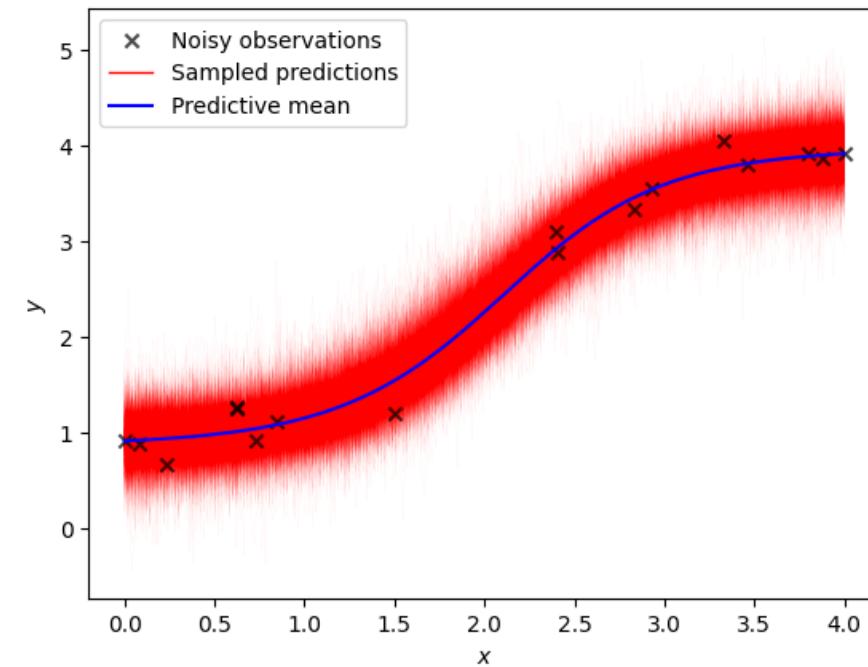
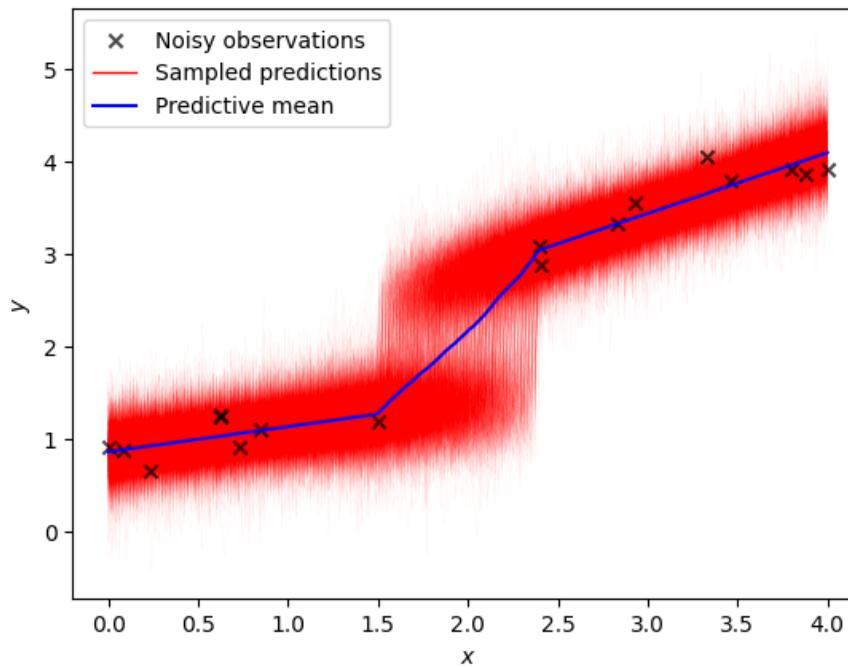
Bayesian Inference



Bayesian Inference



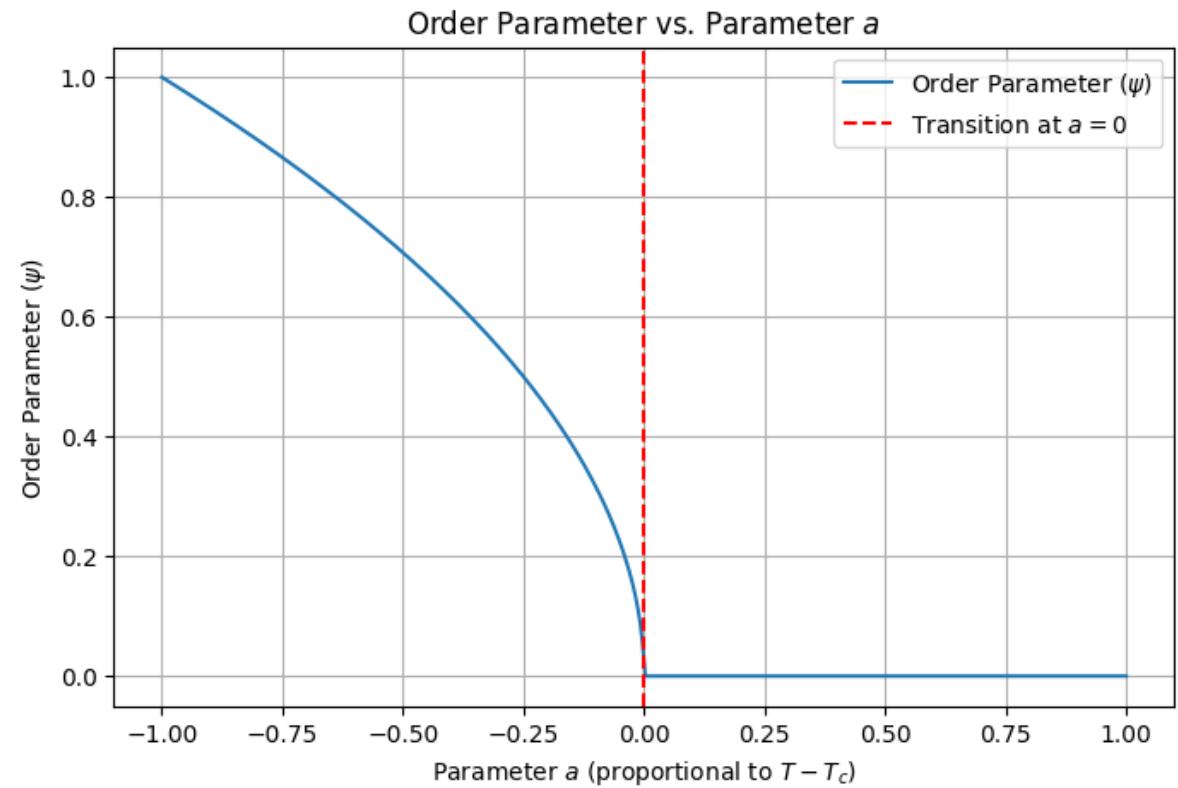
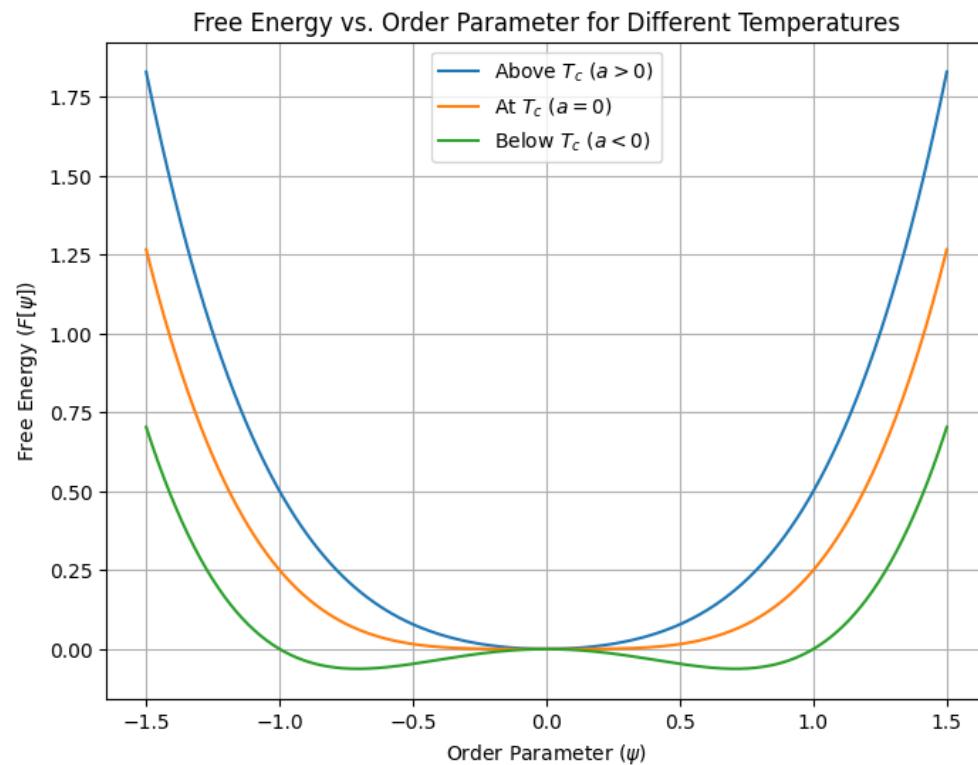
Comparison of models: WAIC



```
1 az.compare(mcmc_all, ic="waic")
```

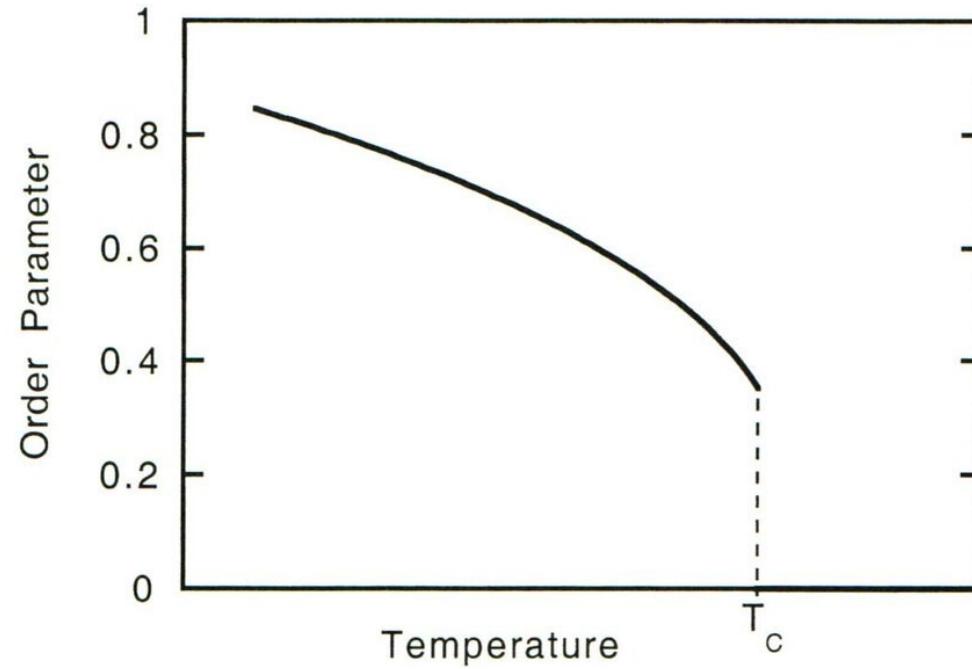
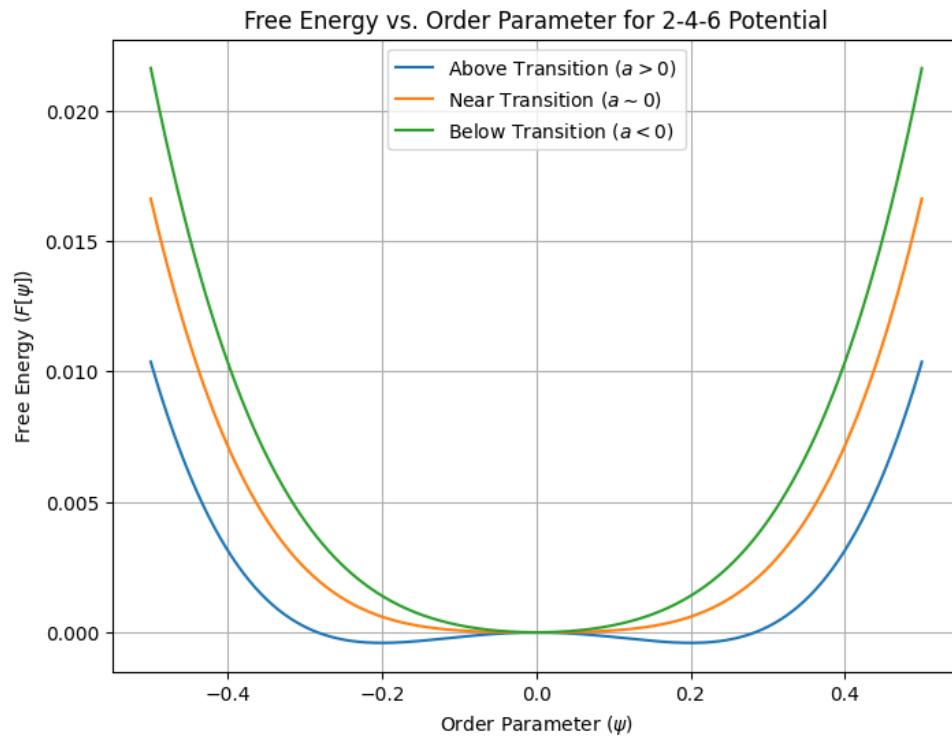
	rank	elpd_waic	p_waic	elpd_diff	weight	se	dse	warning	scale
piecewise	0	1.305878	3.207341	0.000000	5.027393e-01	2.217737	0.000000	True	log
sigmoidal	1	0.845156	4.073488	0.460722	4.972607e-01	2.829716	2.693397	True	log
linear2	2	-6.792783	2.312571	8.098661	0.000000e+00	2.842782	3.265961	True	log
power_law2	3	-7.820722	3.031942	9.126600	0.000000e+00	2.667691	3.113230	True	log
exponential1	4	-9.072673	2.661915	10.378551	0.000000e+00	2.251935	2.767500	True	log
linear1	5	-9.886713	1.539532	11.192591	9.836576e-14	3.207155	3.532880	True	log

Ginzburg-Landau Theory



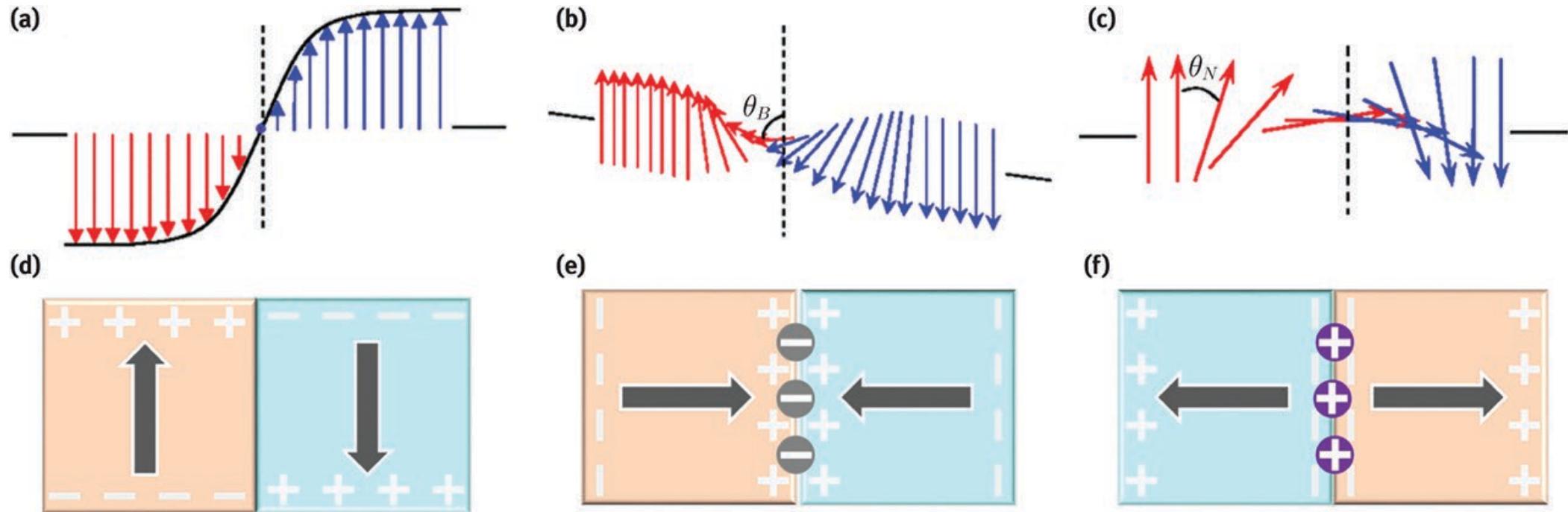
$$F = a(T-T_c) x^2 + b x^4$$

Ginzburg-Landau Theory



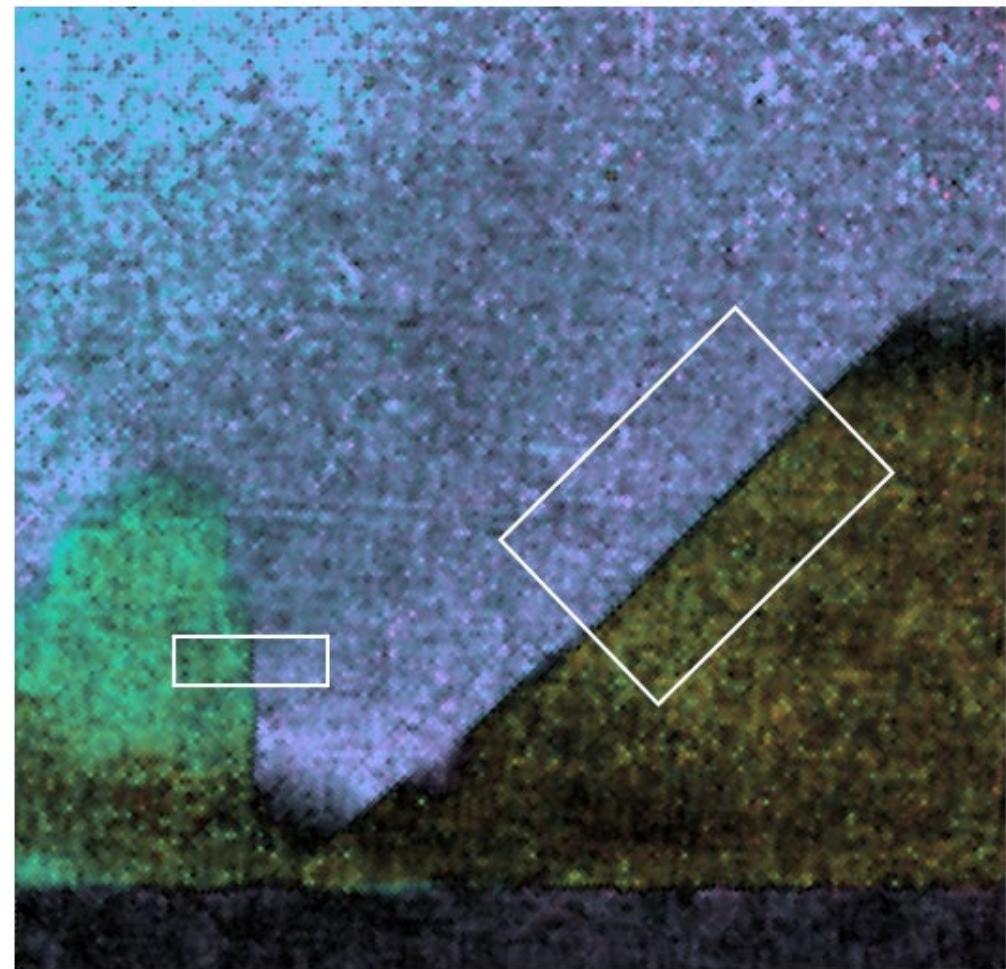
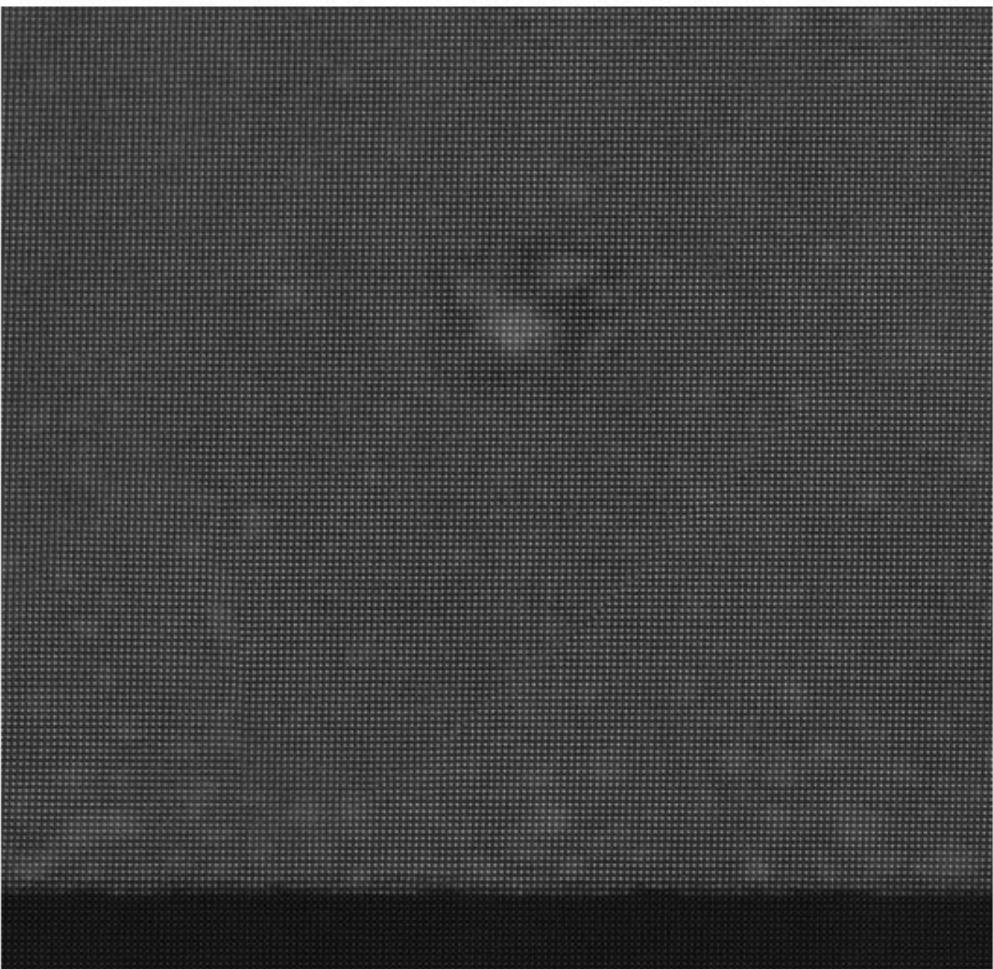
$$F = a(T-T_c) x^2 + b x^4 + c x^6$$

Topological defects and gradient terms

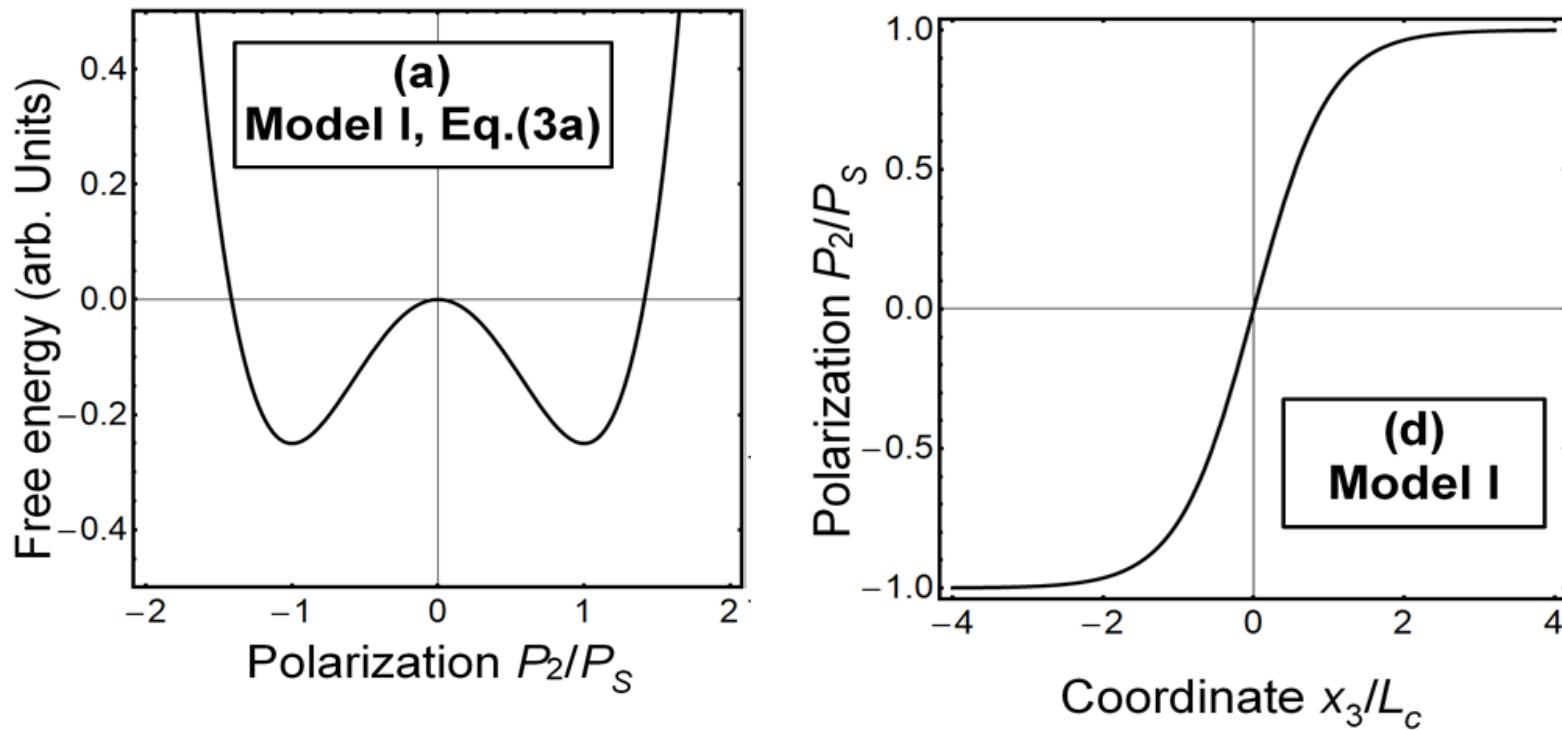


Energy \sim gradient of order parameter

Experiment: Electron Microscopy



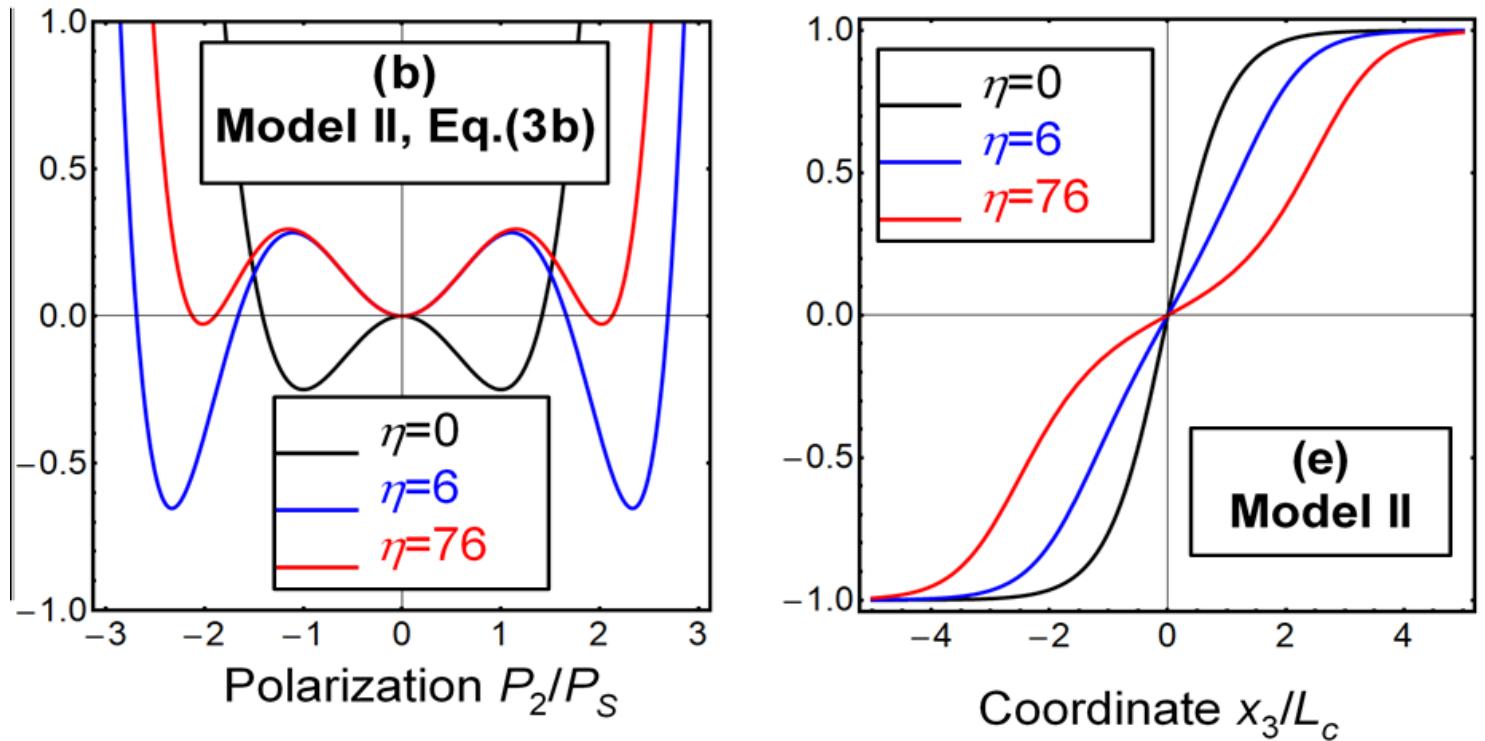
Model: Ferroelectric order parameter



Model 1. For the ferroelectrics with the second order phase transition the order parameter \mathbf{P} across the uncharged domain wall can be found in the one component and one-dimensional approximations. Namely:

$$P_2(x_3) = P_S \cdot \tanh\left[\frac{x_3 - x_0}{L_c}\right], P_1(x_3) = 0, \quad (3a)$$

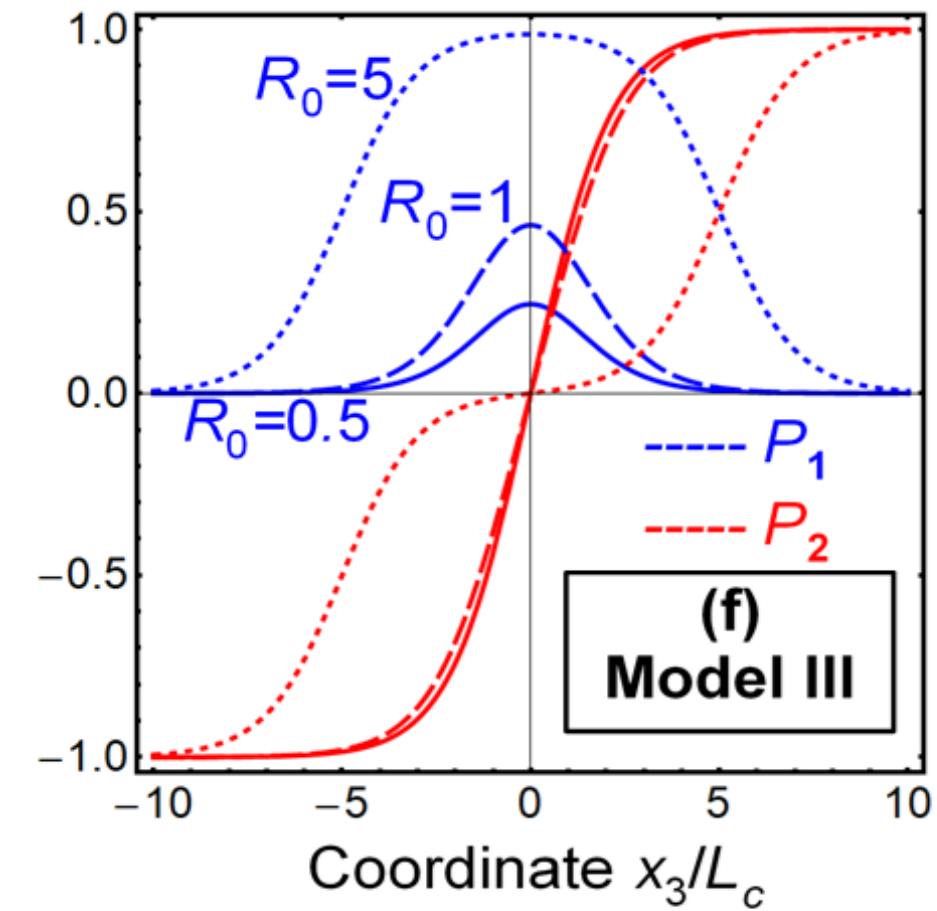
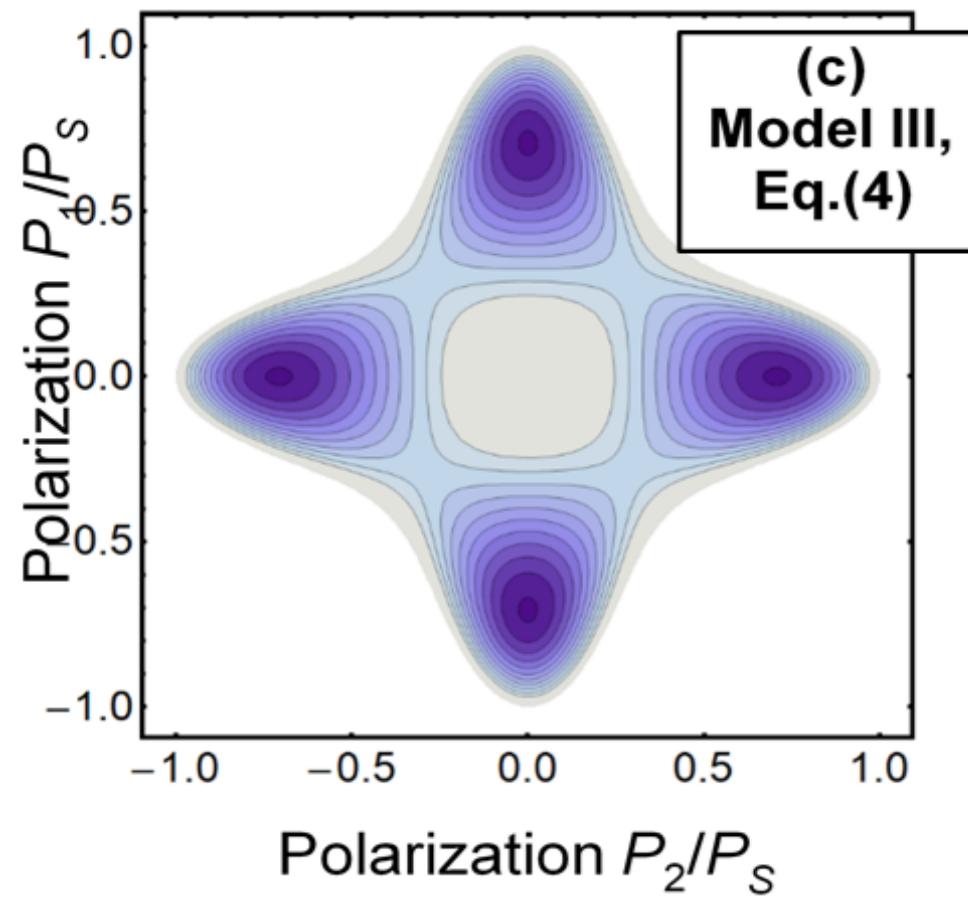
Model: Ferroelectric order parameter



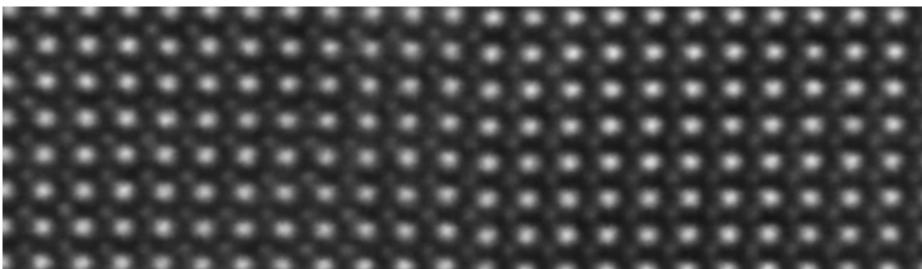
Model 2. For the ferroelectrics with the first order phase transition, the order parameter profile is more complex:

$$P_2(x_3) = \frac{P_S \cdot \sinh[(x_3 - x_0)/L_c]}{\sqrt{\eta + \cosh^2 [(x_3 - x_0)/L_c]}}, P_1(x_3) = 0, \quad (3b)$$

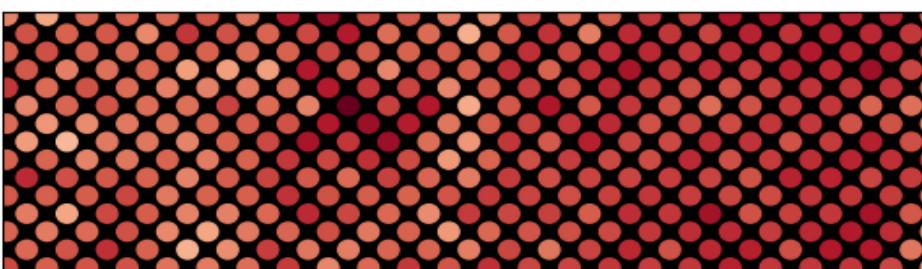
Model: Ferroelectric order parameter



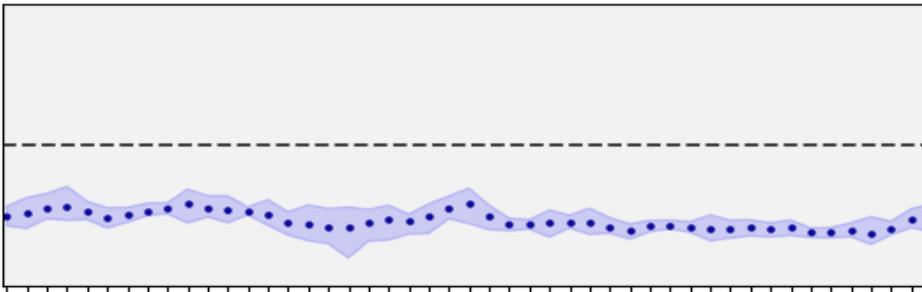
Analysis: Wall Profile



$P \perp$



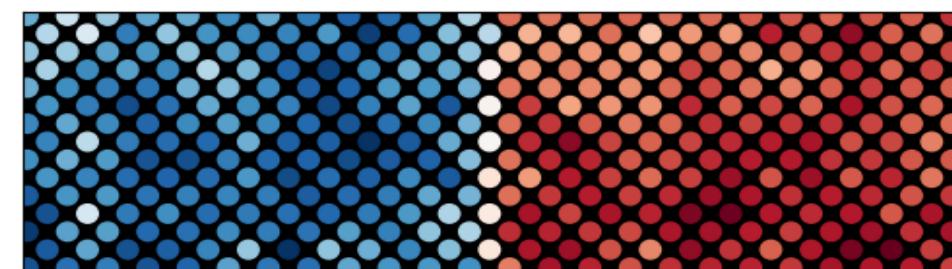
$P \perp$ profile



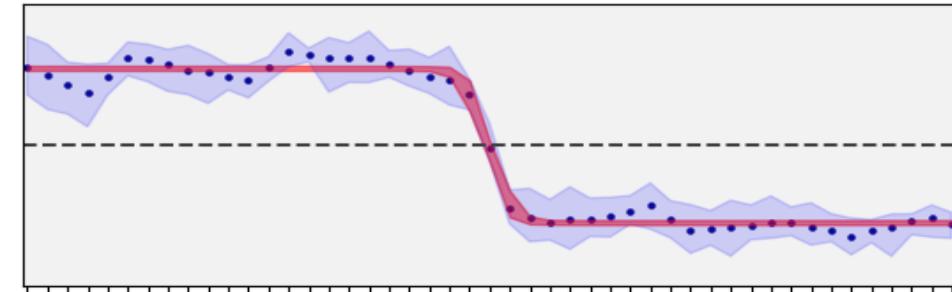
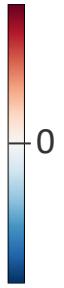
Polar Displacement
Vector



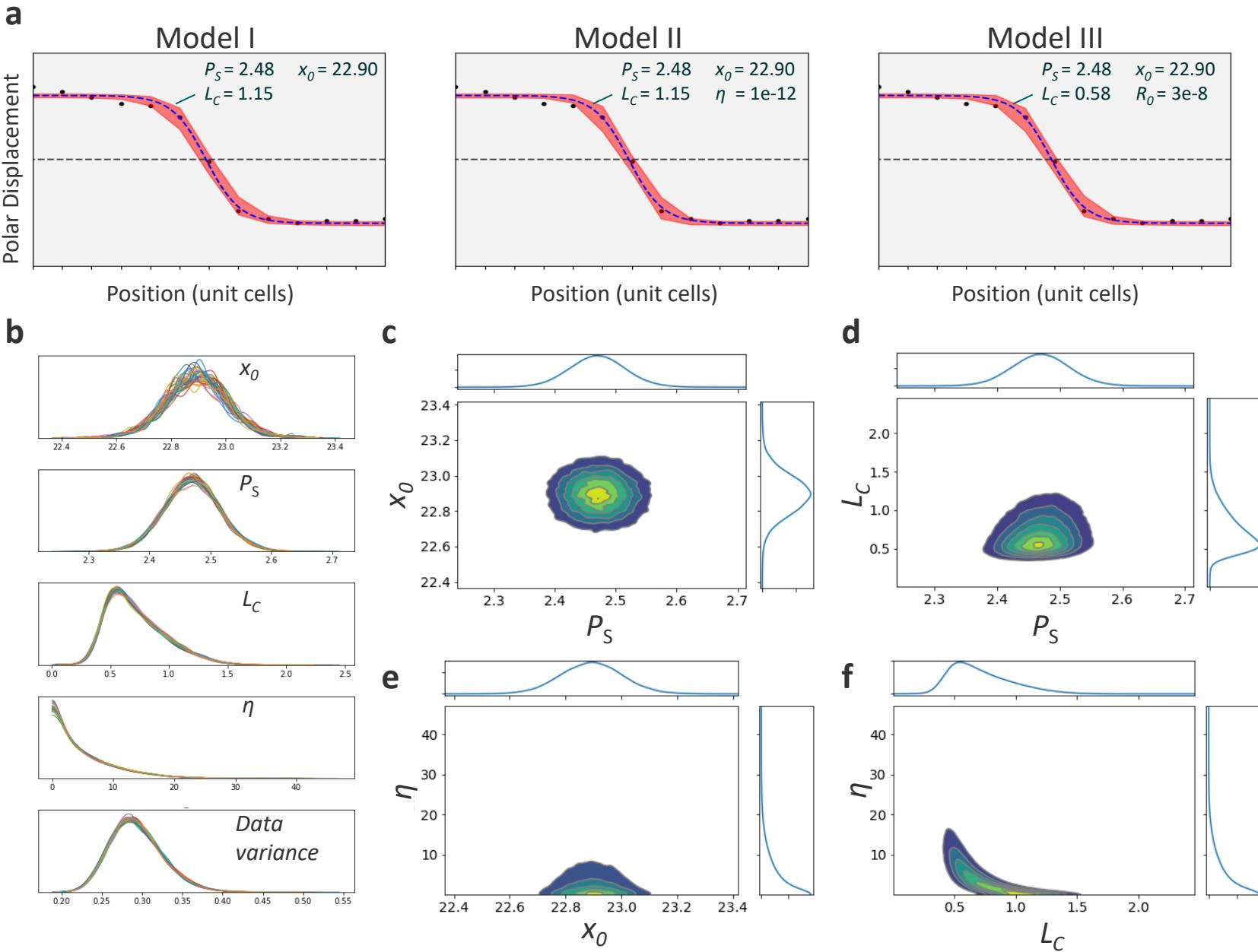
$P \parallel$



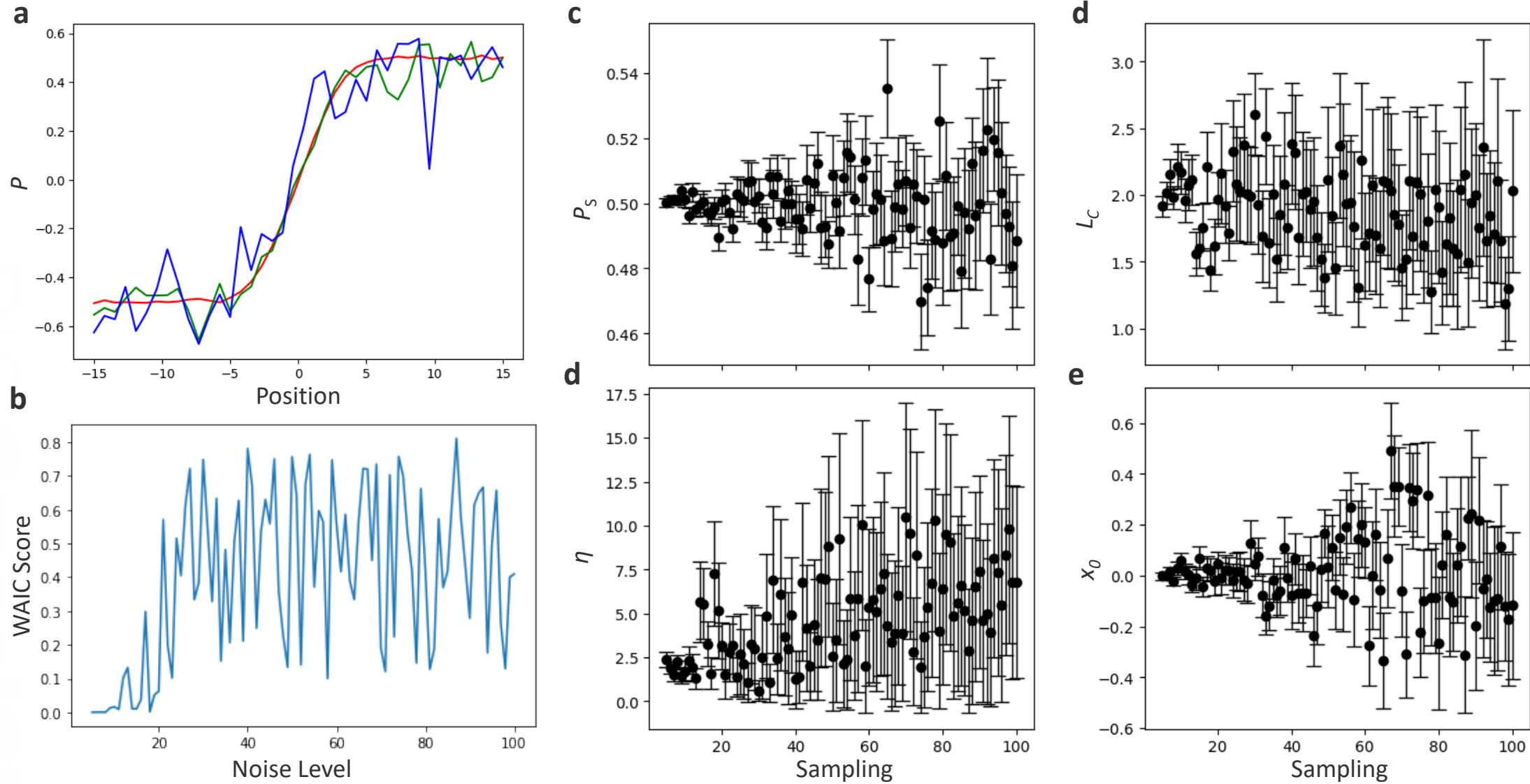
$P \parallel$ profile



Bayesian Inference

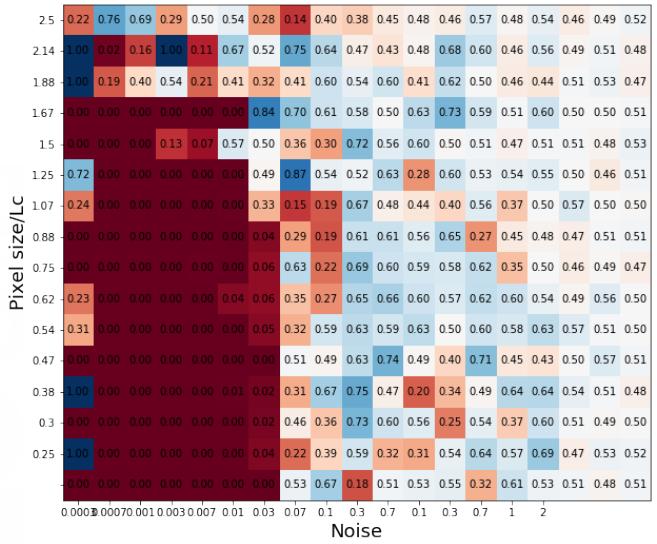


So, what microscope do we need?

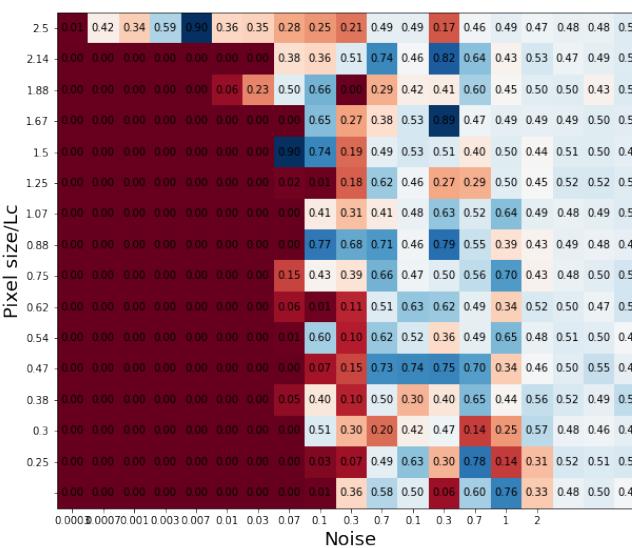


Effect of prior knowledge

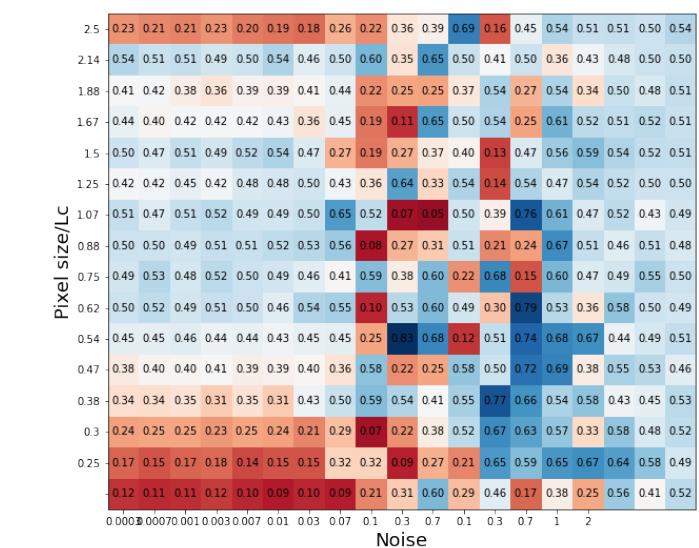
$P_s \sim U(0.4, 0.6), L_c \sim U(0.5, 5.0)$



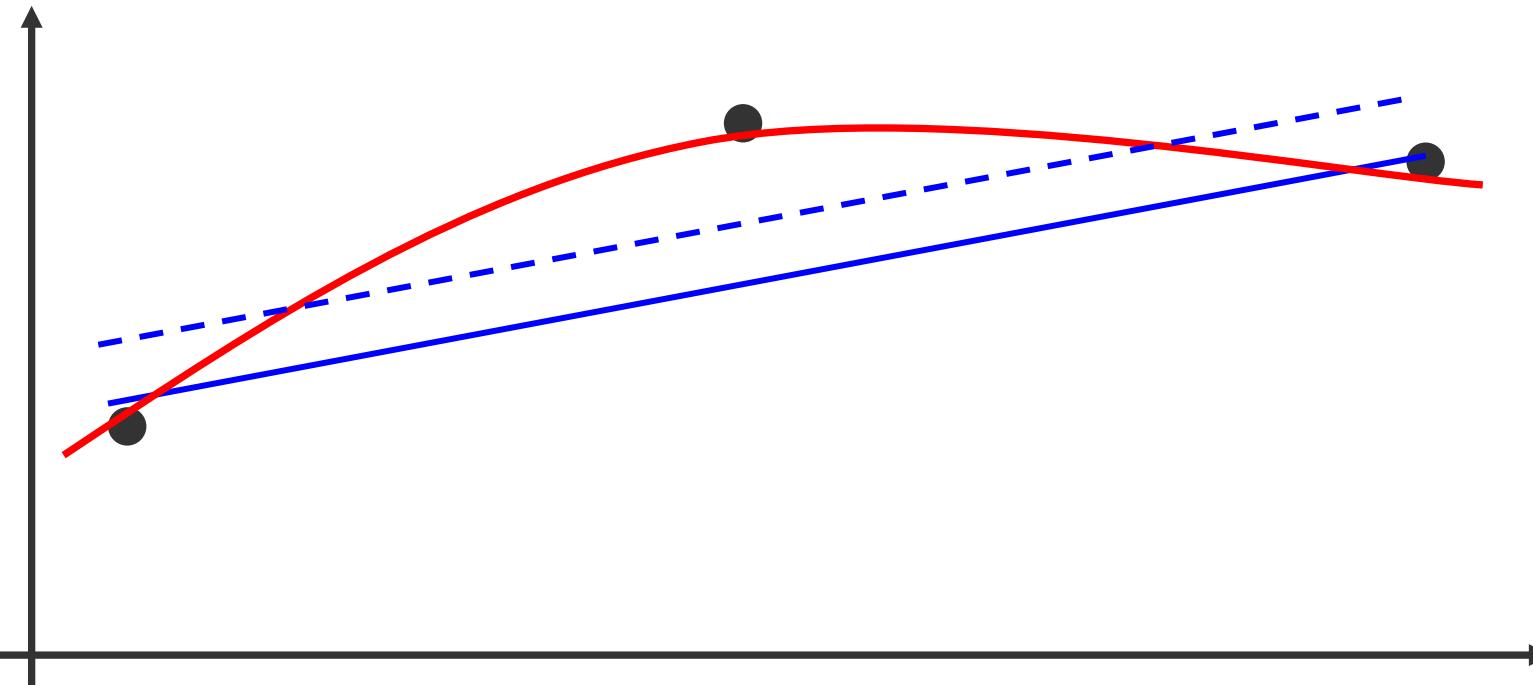
$P_s \sim U(0.49, 0.51), L_c \sim U(1.9, 2.1)$



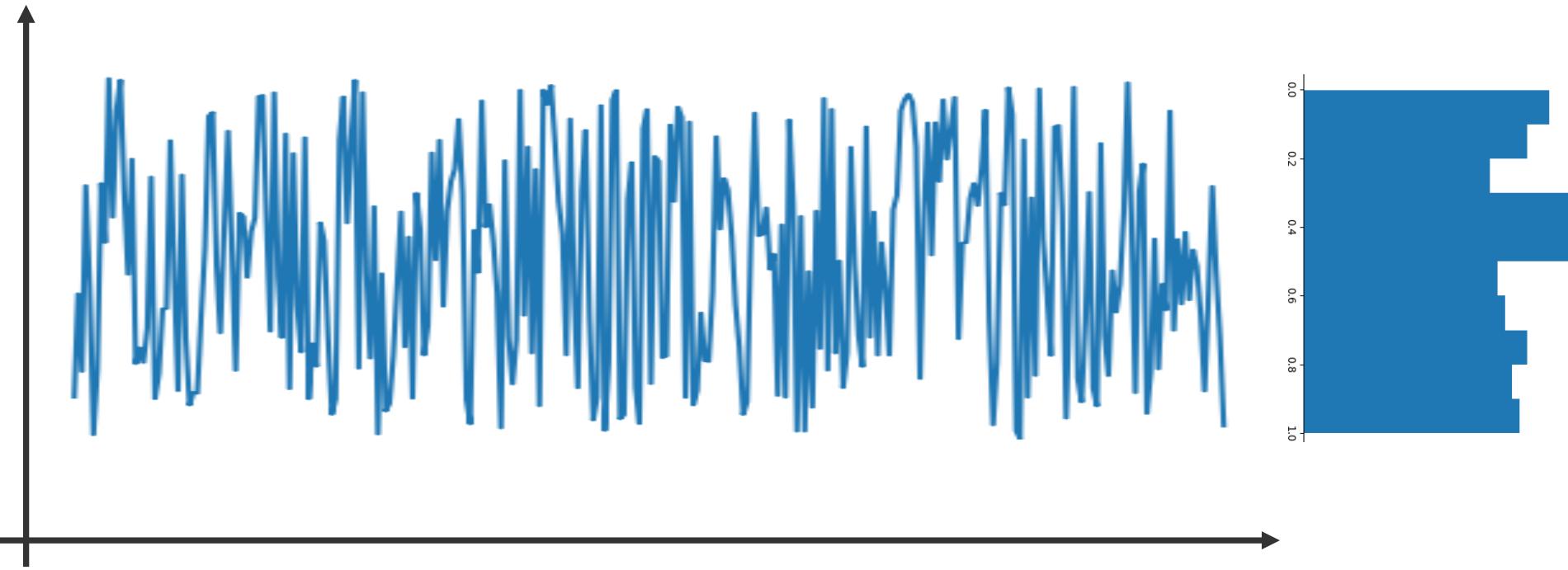
$P_s = 0.7, L_c \sim U(0.5, 5)$



What do we know if we do not know anything?



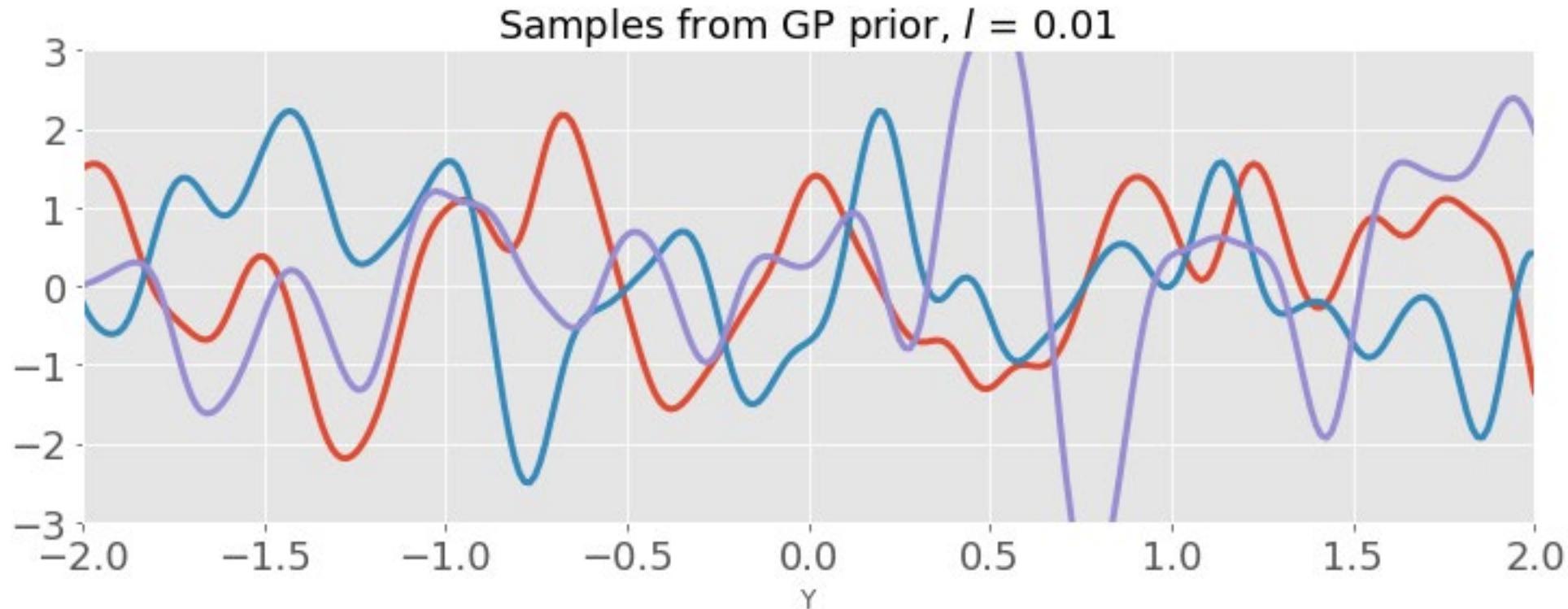
What do we know if we do not know anything?



Gaussian Process Regression

- Covariance matrix determines what type of functions we will allow.

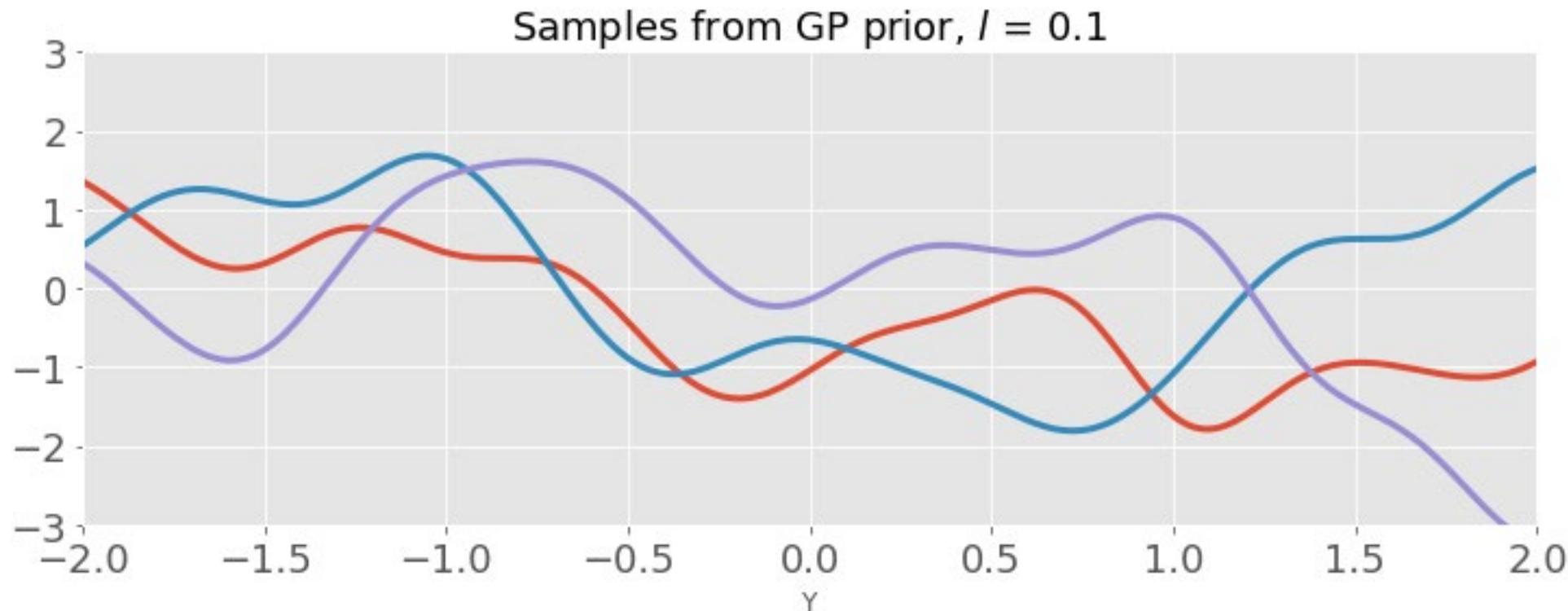
$$k(x, x') = \exp\left(-\frac{1}{2l}(x - x')^2\right)$$



Gaussian Process Regression

- Covariance matrix determines what type of functions we will allow.

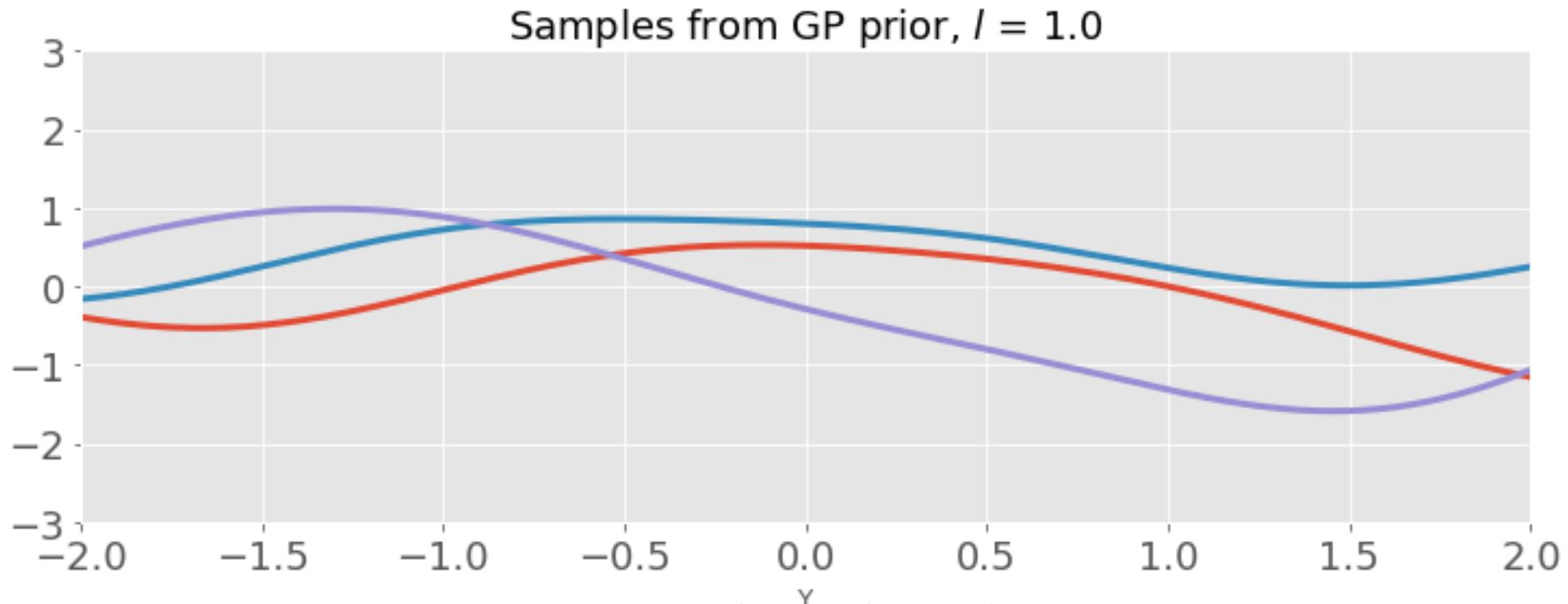
$$k(x, x') = \exp\left(-\frac{1}{2l}(x - x')^2\right)$$



Gaussian Process Regression

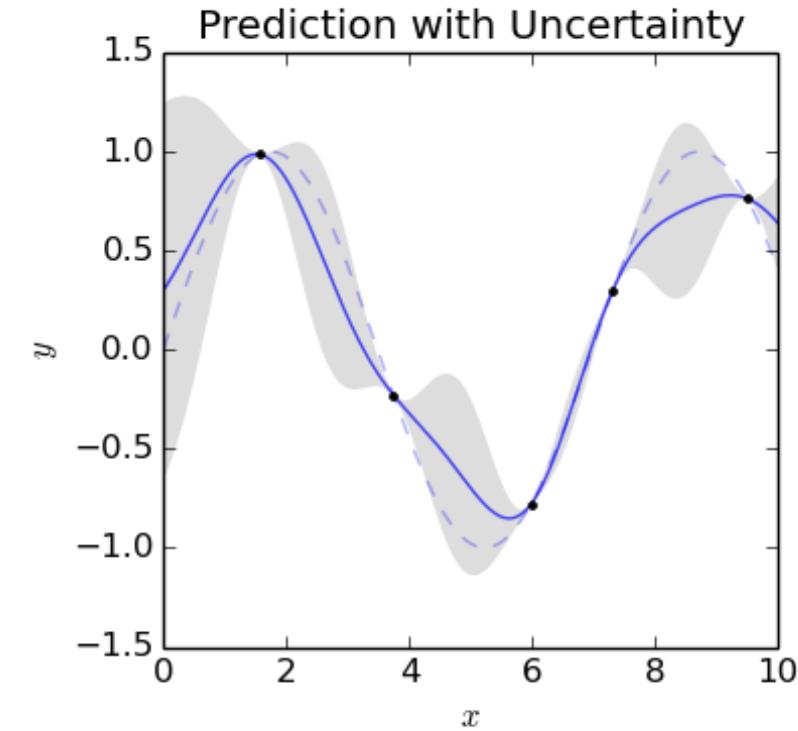
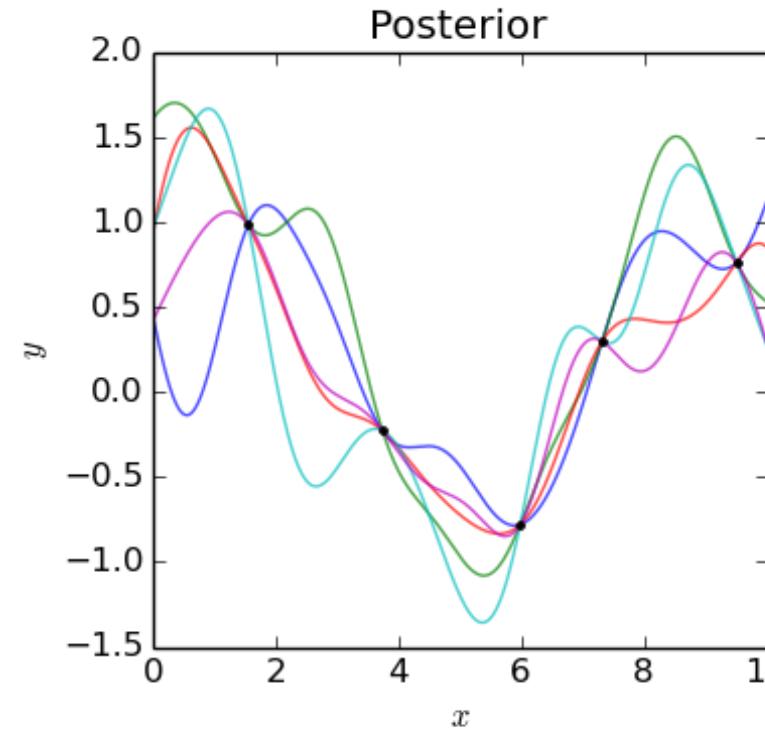
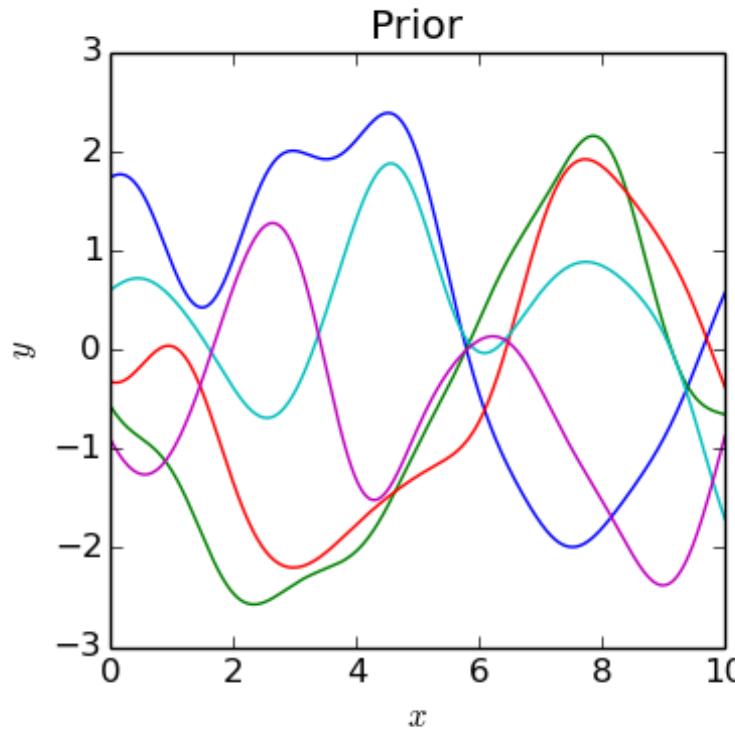
- Covariance matrix (kernel) determines what type of functions we will allow.

$$k(x, x') = \exp\left(-\frac{1}{2l}(x - x')^2\right)$$



l controls the length scale – sort of how far points should be to make them independent of each other.

Gaussian Process Regression



Prior:

Data:

Posterior:

What can the function be before the measurement

Measurement

What can the function be after measurement

Policy:

How do we balance exploration and exploitation (acquisition function)

GP Vocabulary

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

Kernels in Molecular Spaces

Graph Kernels: Since chemical molecules can be represented as graphs with atoms as nodes and bonds as edges, graph kernels are a natural choice. Graph kernels measure the similarity between graphs and can be used in GPs to model molecular properties. Examples include the Random Walk Kernel, Weisfeiler-Lehman (WL) Kernel, and Graphlet Kernel.

Molecular Fingerprint Kernels: Molecular fingerprints are binary vectors representing the presence or absence of certain substructures within a molecule. Kernels can be defined on these fingerprints, such as the Tanimoto Kernel (also known as the Jaccard Index), which measures the similarity between two fingerprint vectors. This approach is particularly popular in cheminformatics for its interpretability and efficiency.

Spectral Kernels: Spectral methods can be used to define kernels on the spectrum (eigenvalues) of the Laplacian matrix of molecular graphs. These kernels can capture the topological and spectral properties of molecules, offering a powerful way to model molecular similarities.

Kernels in Molecular Spaces

Gaussian Molecular Kernels: This type of kernel measures the similarity between molecules based on Gaussian-shaped functions centered on atoms. The overlap between these Gaussian functions for different molecules can be used to compute a kernel value, incorporating both structural and spatial information.

Physicochemical Property Kernels: Kernels can be designed based on the physicochemical properties of molecules, such as hydrophobicity, charge distribution, or molecular weight. By defining a similarity measure based on the distance between these property vectors of different molecules, one can construct a kernel that reflects the similarities in physical and chemical properties.

Deep Learning-Based Kernels: With the rise of deep learning, kernels can also be derived from the embeddings or latent spaces generated by neural networks trained on molecular data. For instance, a neural network can be trained to generate molecular embeddings, and a kernel can be defined as the dot product or cosine similarity between these embeddings.

Colab