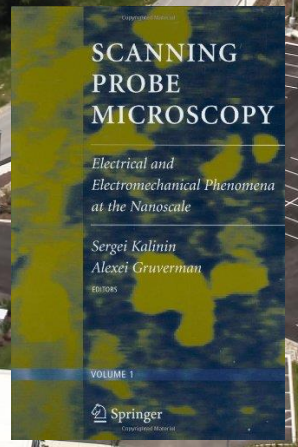
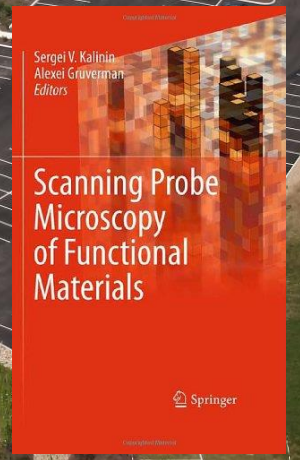
# 2002 - 2022

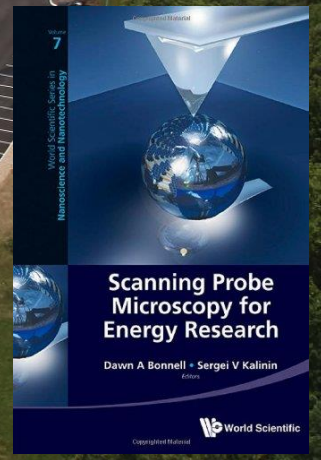
## Since 2023

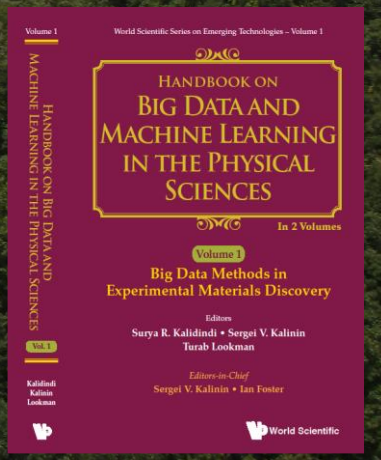
## 2022 -2023

amazon



# Machine Learning and Automated Experiment

1. Machine learning is now everywhere…. But sometimes it is difficult to see it in the lab!

2. Applying ML in experimental sciences can be a very daunting proposition:
   a. Understand the problem
   b. Know (some) ML
   c. Develop code: from prototype to operationalization
   d. Implement on the working instrument
   e. Connect to the cloud
   f. Understand the results

3. The code base and libraries can change (TF/Keras -> PyTorch -> Jax -> ?)

4. Basic concepts are often incomprehensible

# Therefore…

1. We are not going to try to learn all aspects of it. ML in domain/experiment is always a teamwork.

2. The first step is to define your problem – what is that you want to accomplish?

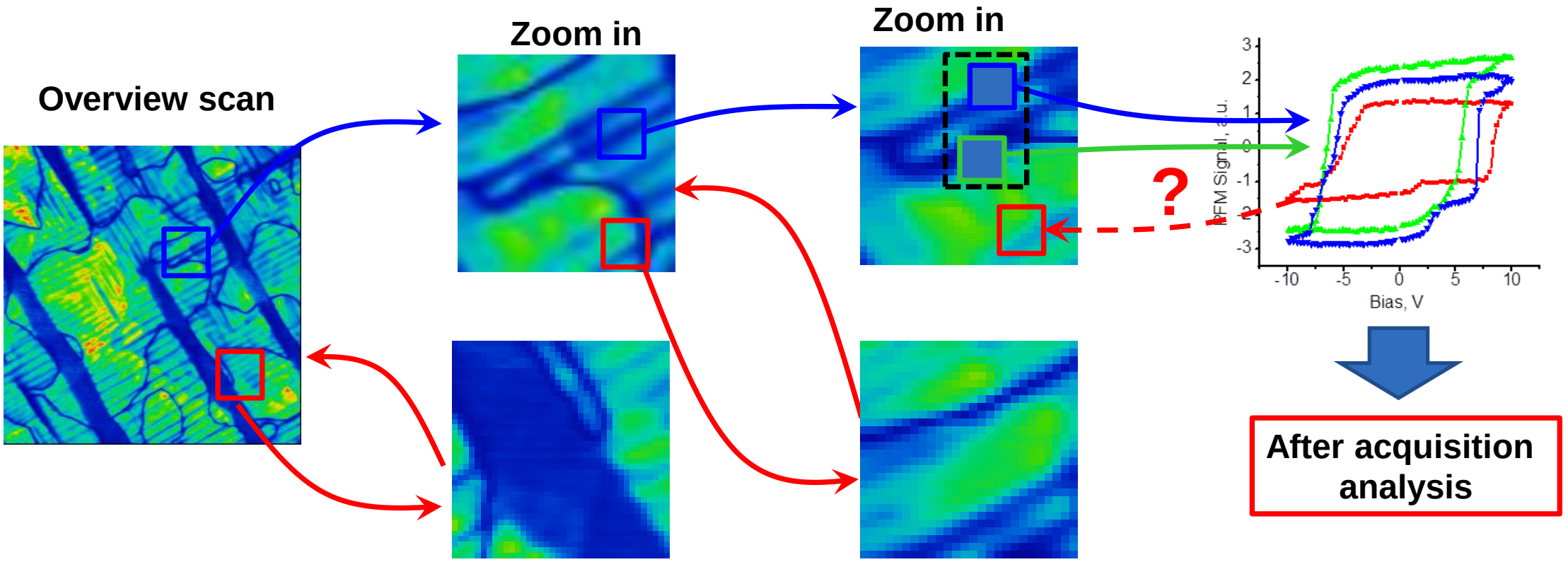
3. The second step is define your hyper language – what are the things that you know how to do (or can learn to do)?

4. The third step is to work backward from your problem and define the workflow in terms of your hyper language. Do you even need machine learning to solve it?

5. For ML, knowing code is (in some sense) secondary. However, it is critical to understand HOW it works.

# Workflows in Scanning Probe Microscopy

Workflow plane

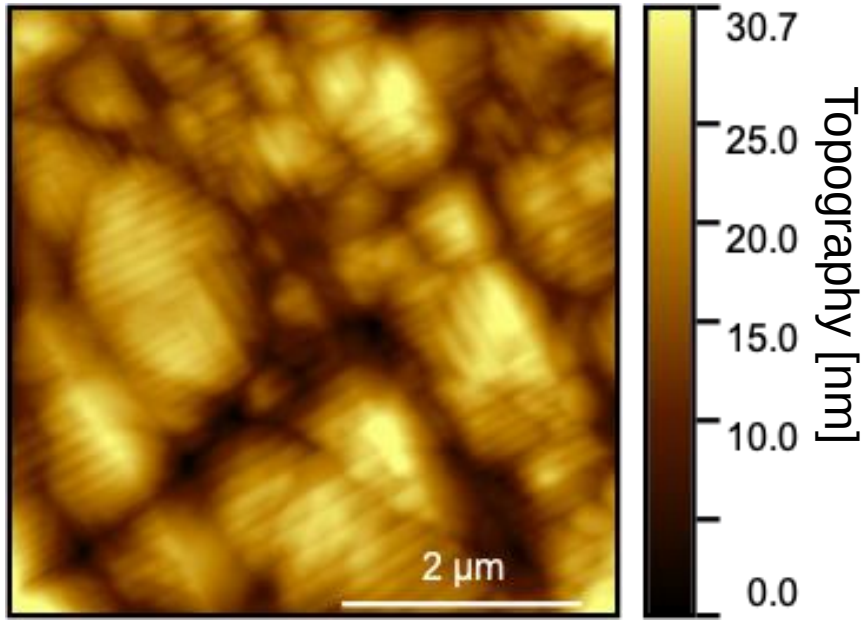


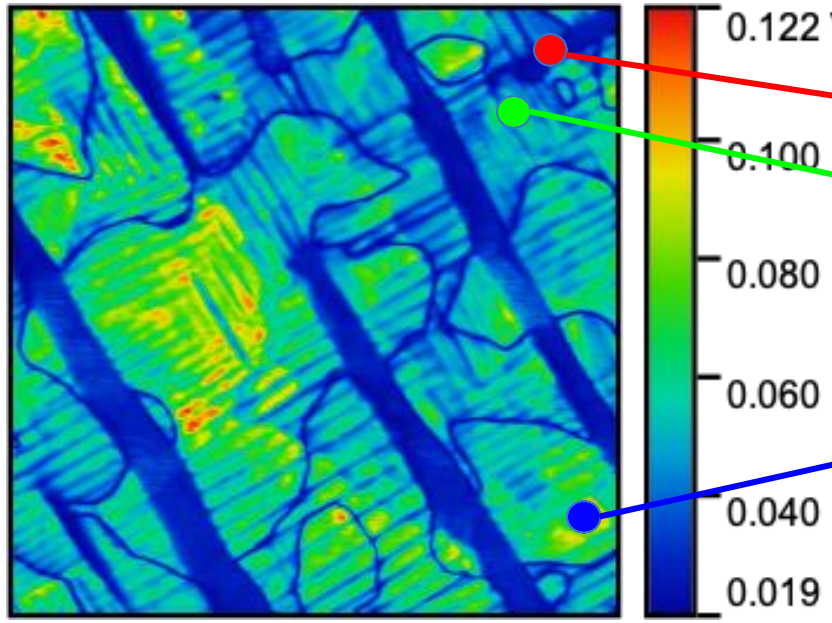

Instrument plane

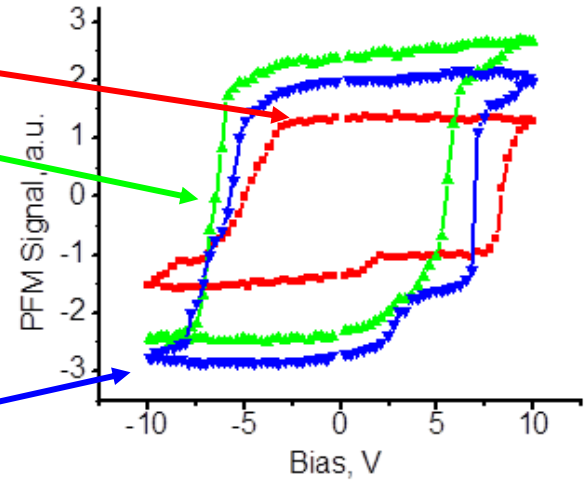
**Minimal instruction set control language**

| Tune microscope | Initiate scan (parameters) | Position probe (x,y,t) | Initiate spectrum (parameters) | Stream data |
|---|---|---|---|---|

# Single step workflow

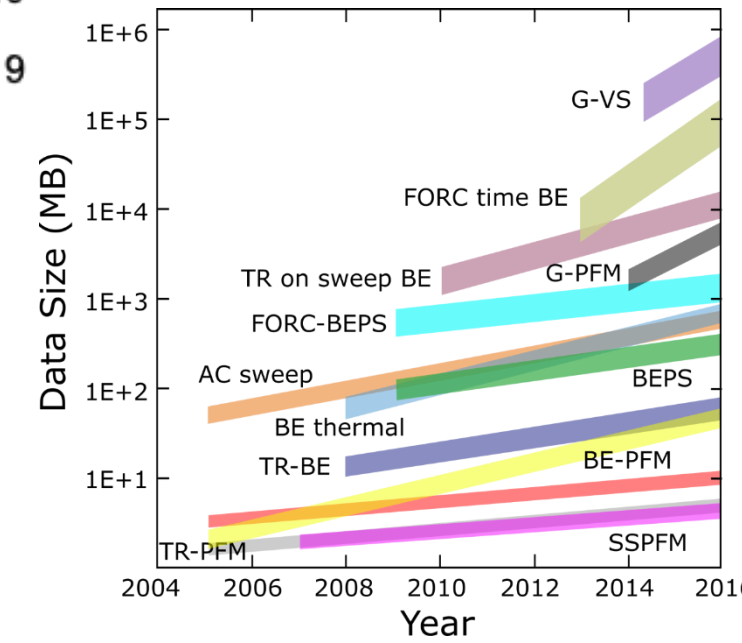


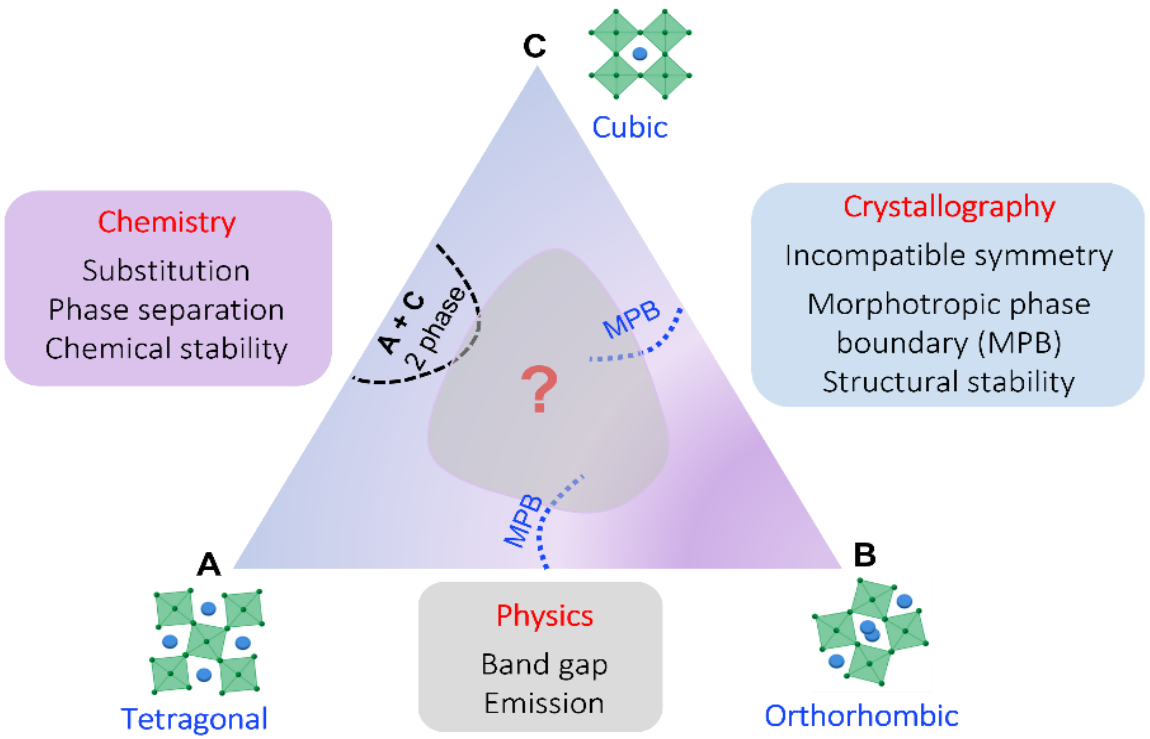



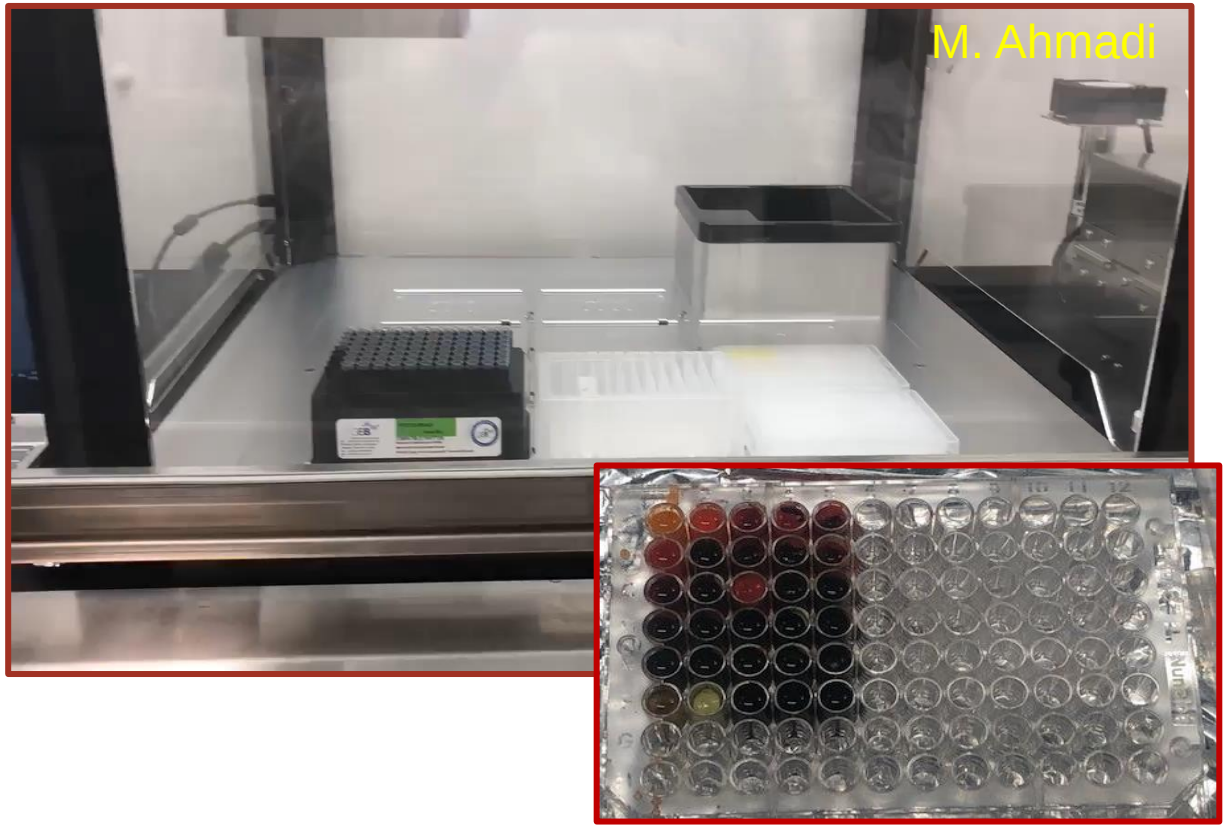

- Interesting functionalities are expected at the certain elements of domain structure
- We can guess some; we have to discover others

- **Can we run experiment so that we either explore**
  - **only selected regions,**
  - **discover new functionalities, or**
  - **seek functionalities we want?**



# Why synthesis (or theory)?

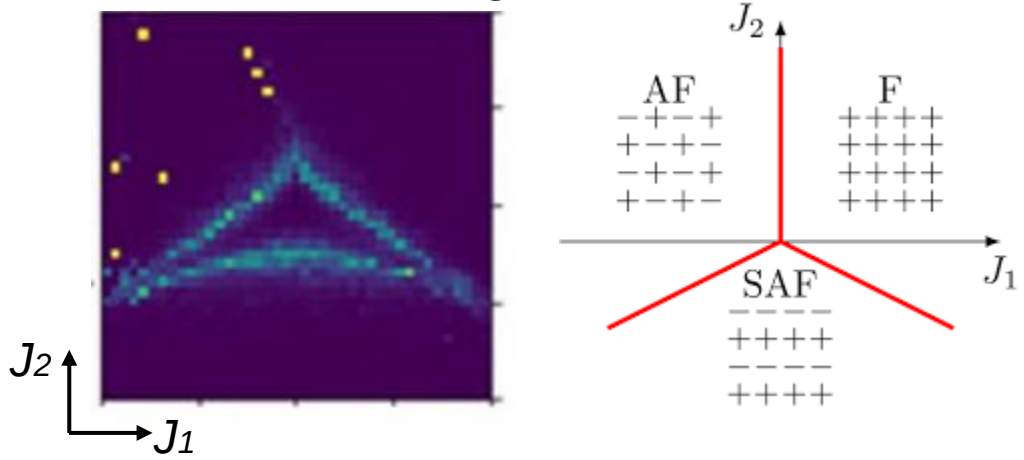



- Automated synthesis in its simplest form requires some way to navigate phase diagrams
- In more complex form, processing space.
- Ideally, incorporate physical knowledge

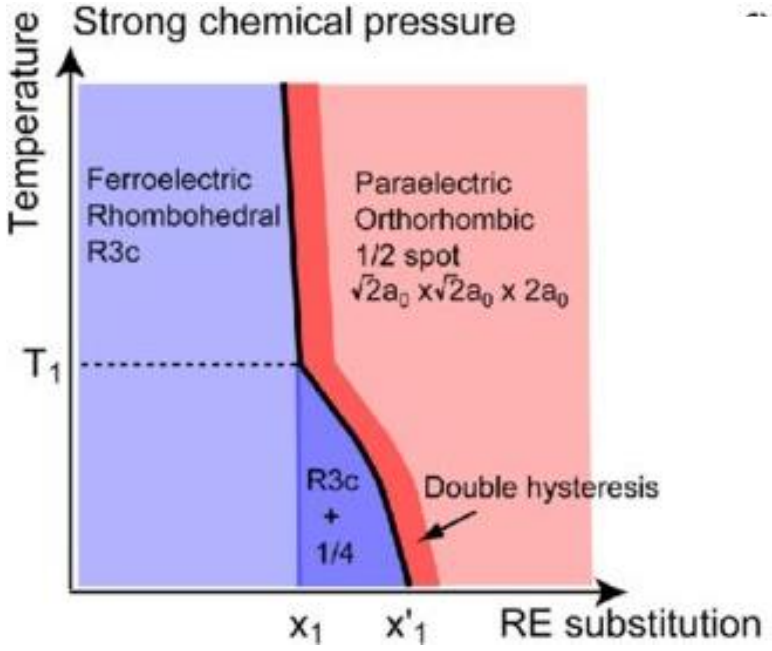
- Similar problem - theory

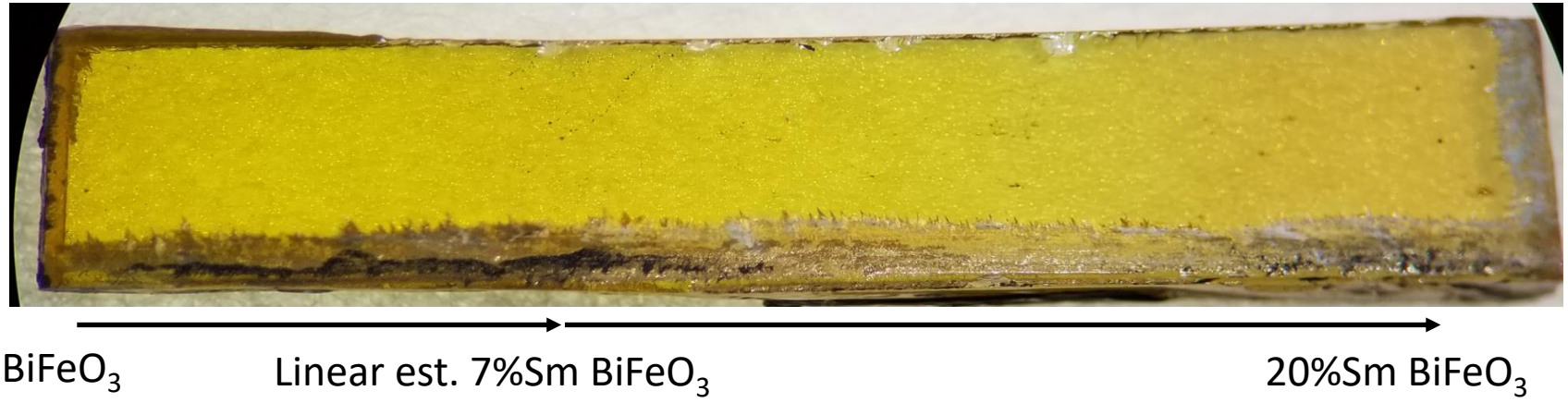
**Ising model**

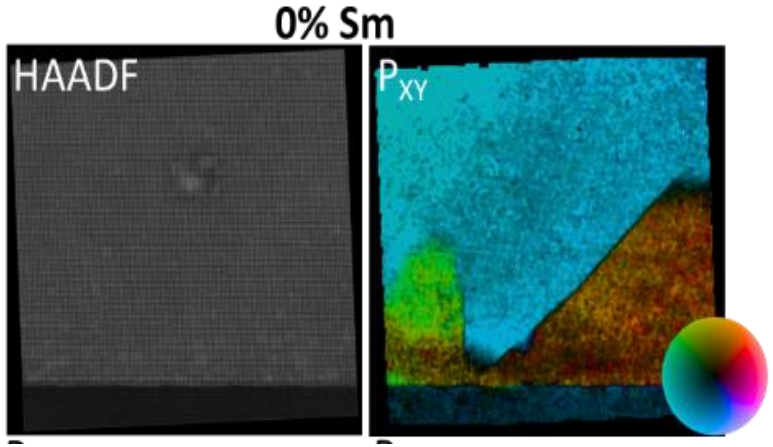


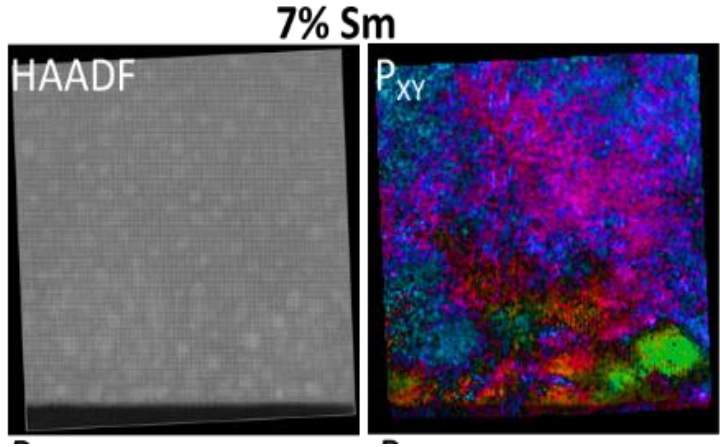
# Combinatorial library

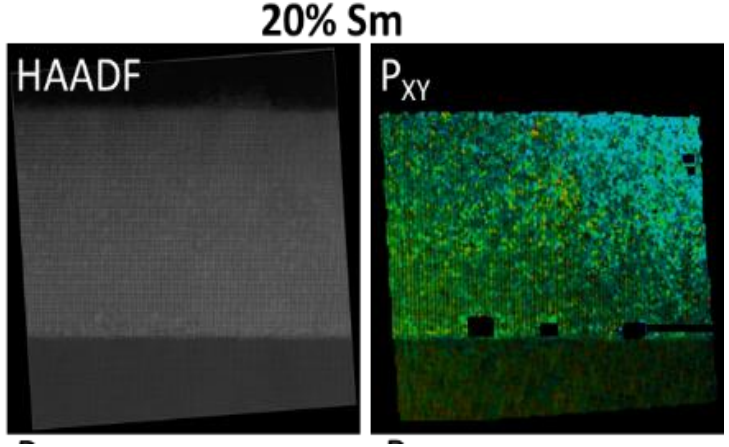
Sample by I. Takeuchi, UMD
Phase diagram by N. Valanoor et al.



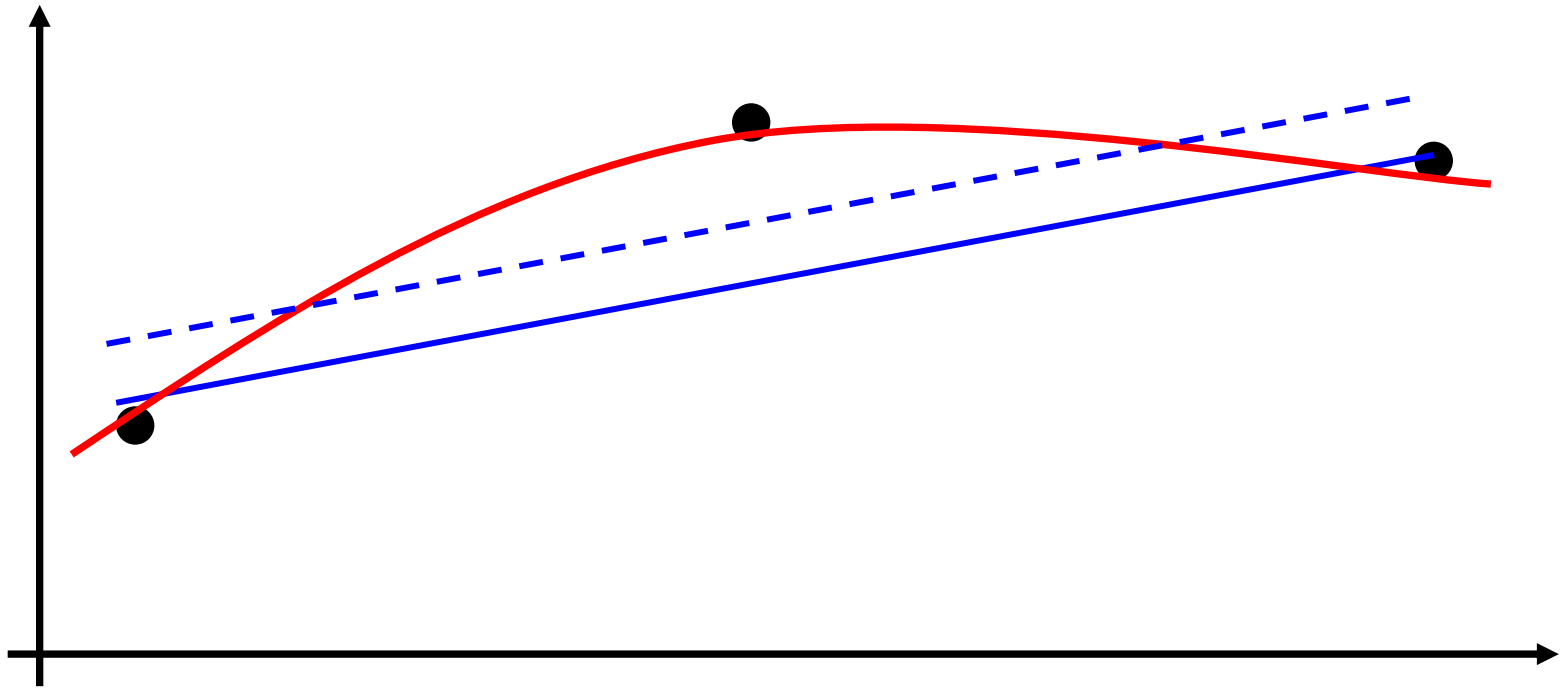




arXiv:2004.11817

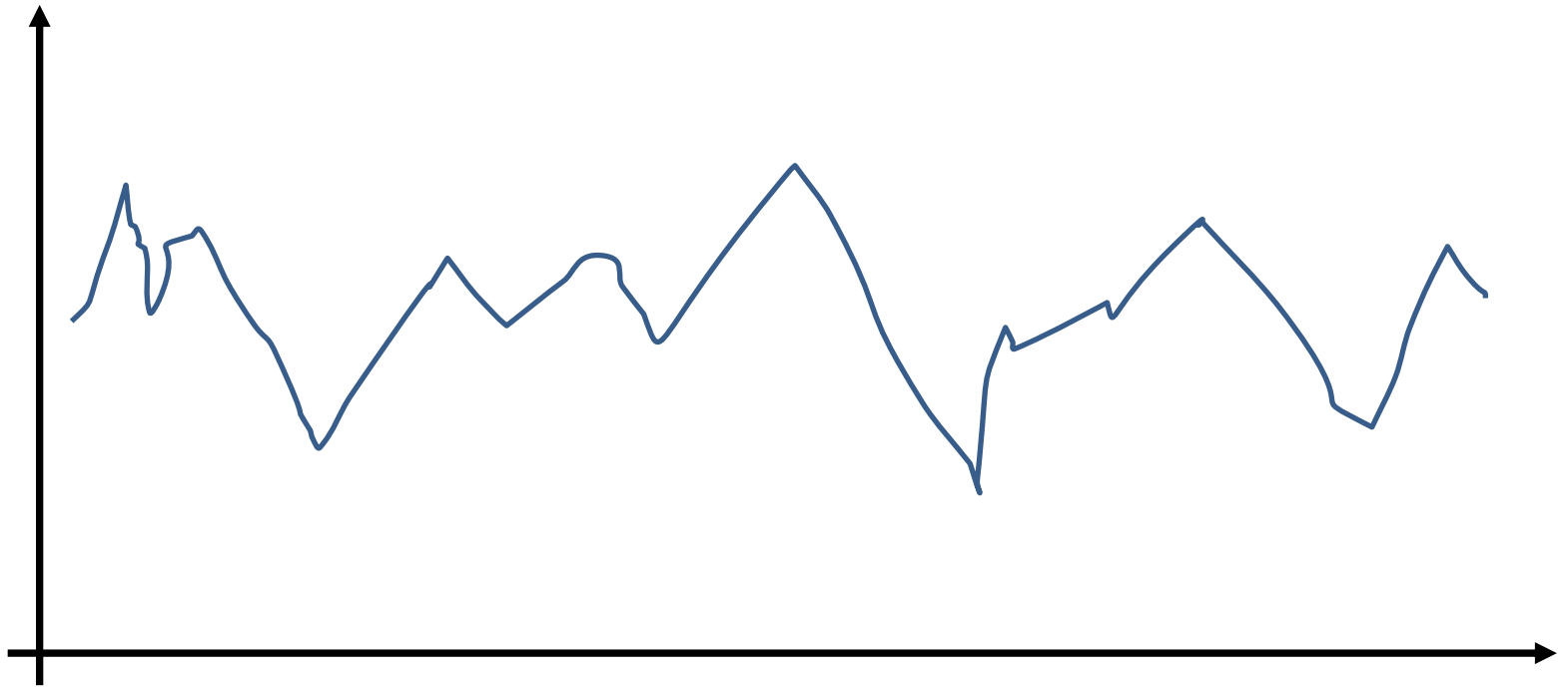
# Automated Experiment:

… with John Snow priors…

# What do we know if we do not know anything?



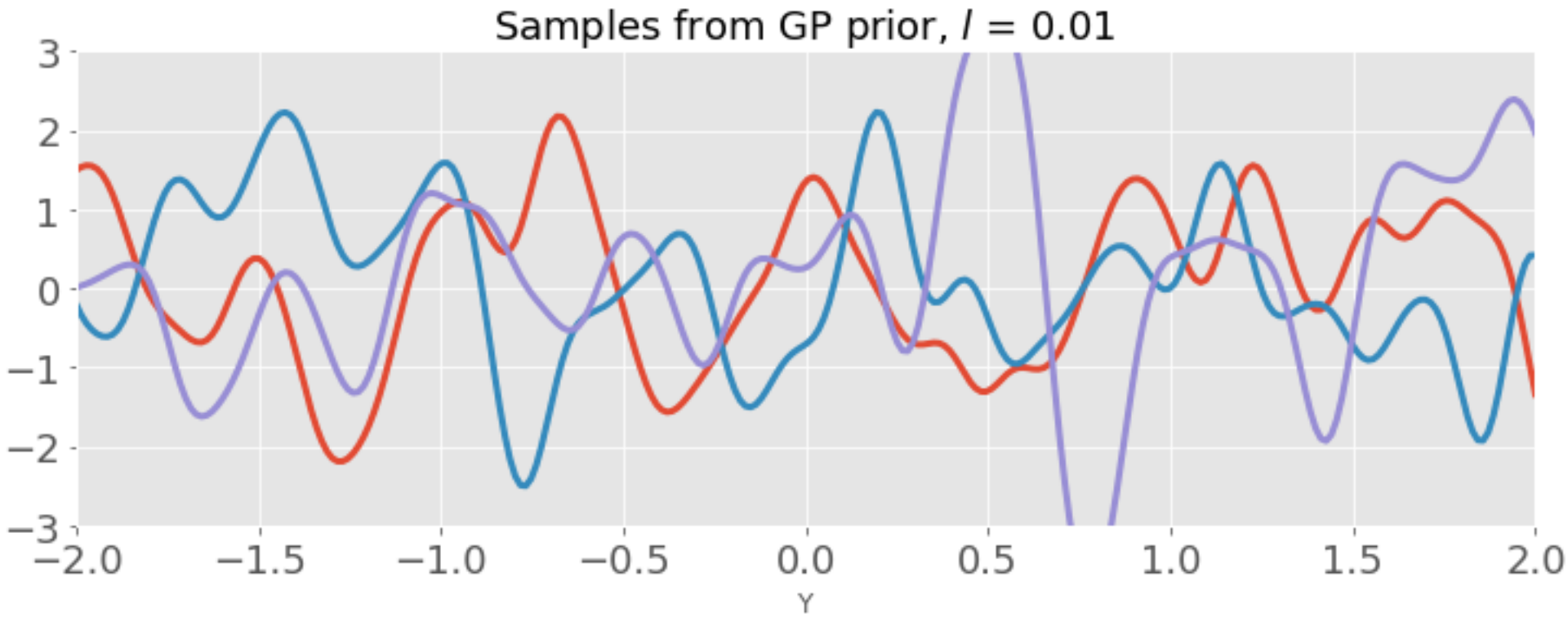
# What do we know if we do not know anything?



# Gaussian Process Regression

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

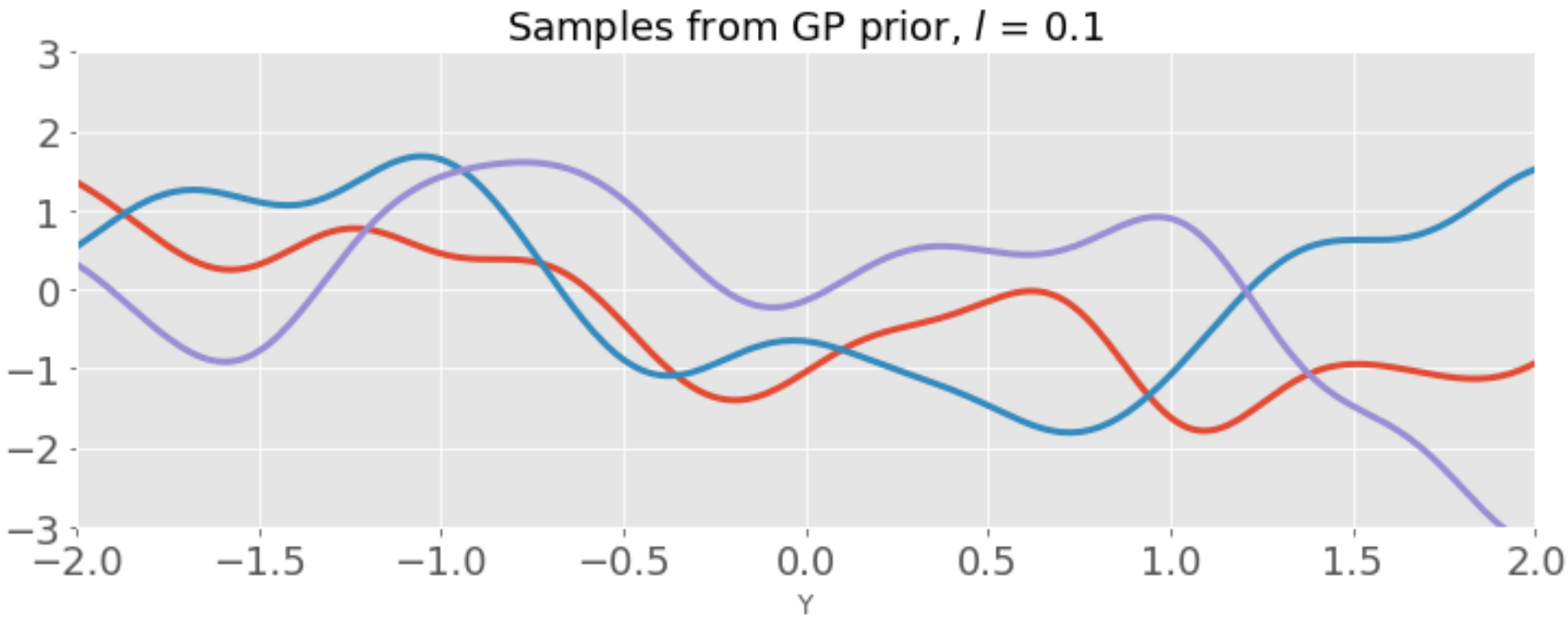
$$\mathrm{k}(\mathrm{x}, \mathrm{x}') = \exp\left(-\frac{1}{2l}(x - x')^2\right)$$



# Gaussian Process Regression

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

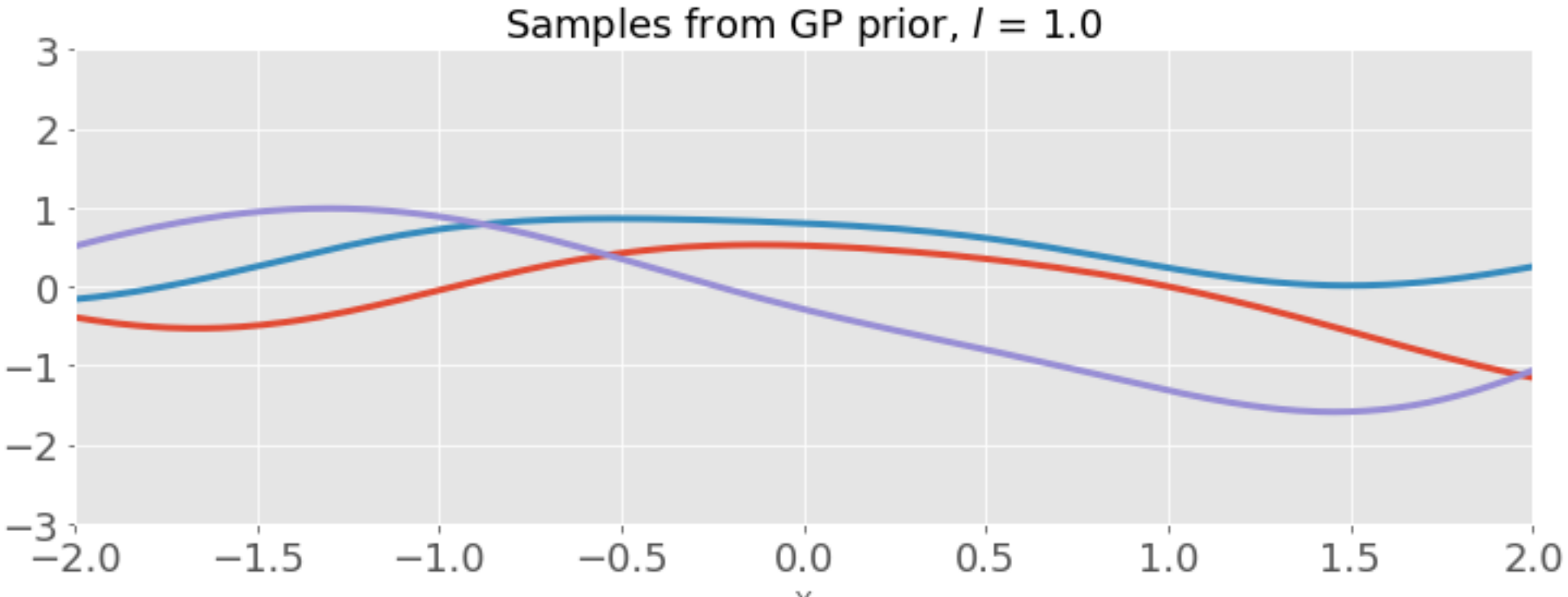
$$\mathrm{k}(\mathrm{x}, \mathrm{x}') = \exp\left(-\frac{1}{2l}(x - x')^2\right)$$



# Gaussian Process Regression

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

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




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