Active Machine Learning with Uncertainties and Gradients:

current state and plan ahead

Khachik Sargsyan

Eric Hermes, Habib Najm, Judit Zador

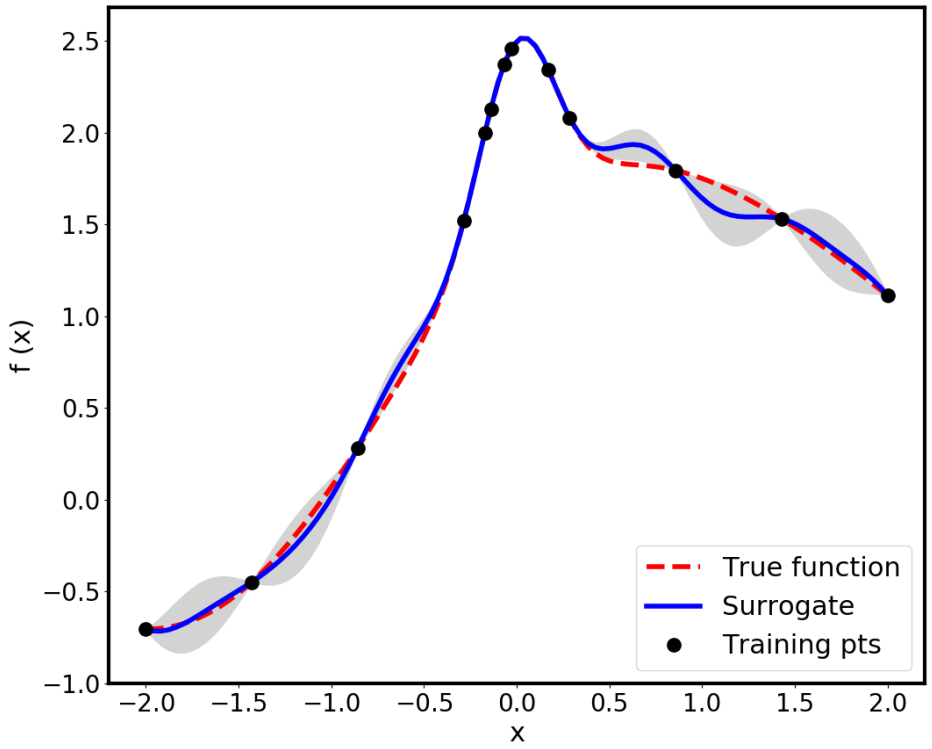
ECC Project Meeting, Livermore, CA

Oct 28, 2019

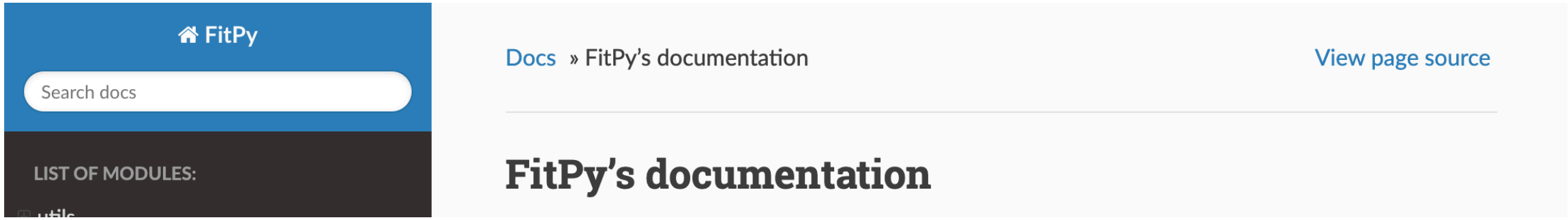
Exascale Catalytic Chemistry

Target: develop capability to incorporate surrogates in PES search mechanisms

- Surrogate = Proxy = Metamodel = Response surface
- Supervised Machine Learning (Surrogate construction = Training)
- Given (x,y) pairs, construct an approximation for y=f(x)
  - x is fingerprint/descriptor
  - y is the energy
- Key requirements:
  - Fast training procedure
  - Surrogate needs to be cheap to evaluate
  - High-dimensionality
  - Respect physics



FitPy: Lightweight Python library at the intersection of UQ and ML



(until recently) Wrapper to

UQTk

- Polynomial chaos:
  - Least-squares (LSQ)
  - Bayesian compressive sensing (BCS)
- Low-rank tensors (LRT)
- Radial basis functions (RBF)
- Gaussian process (GP)

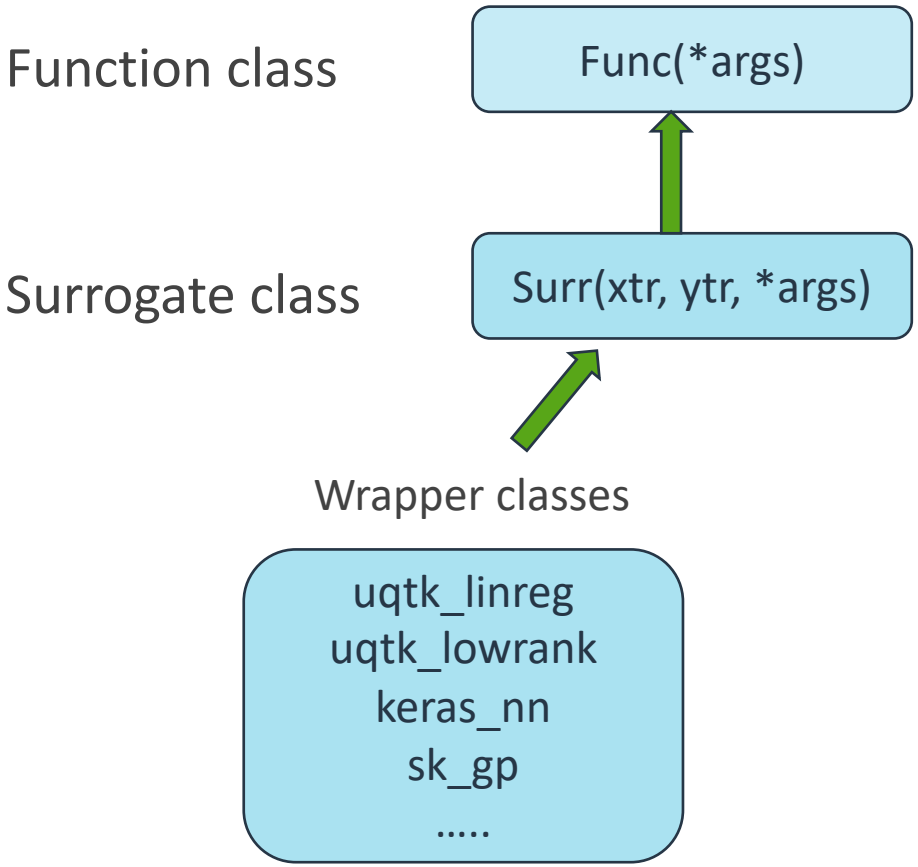
scikit learn

Keras

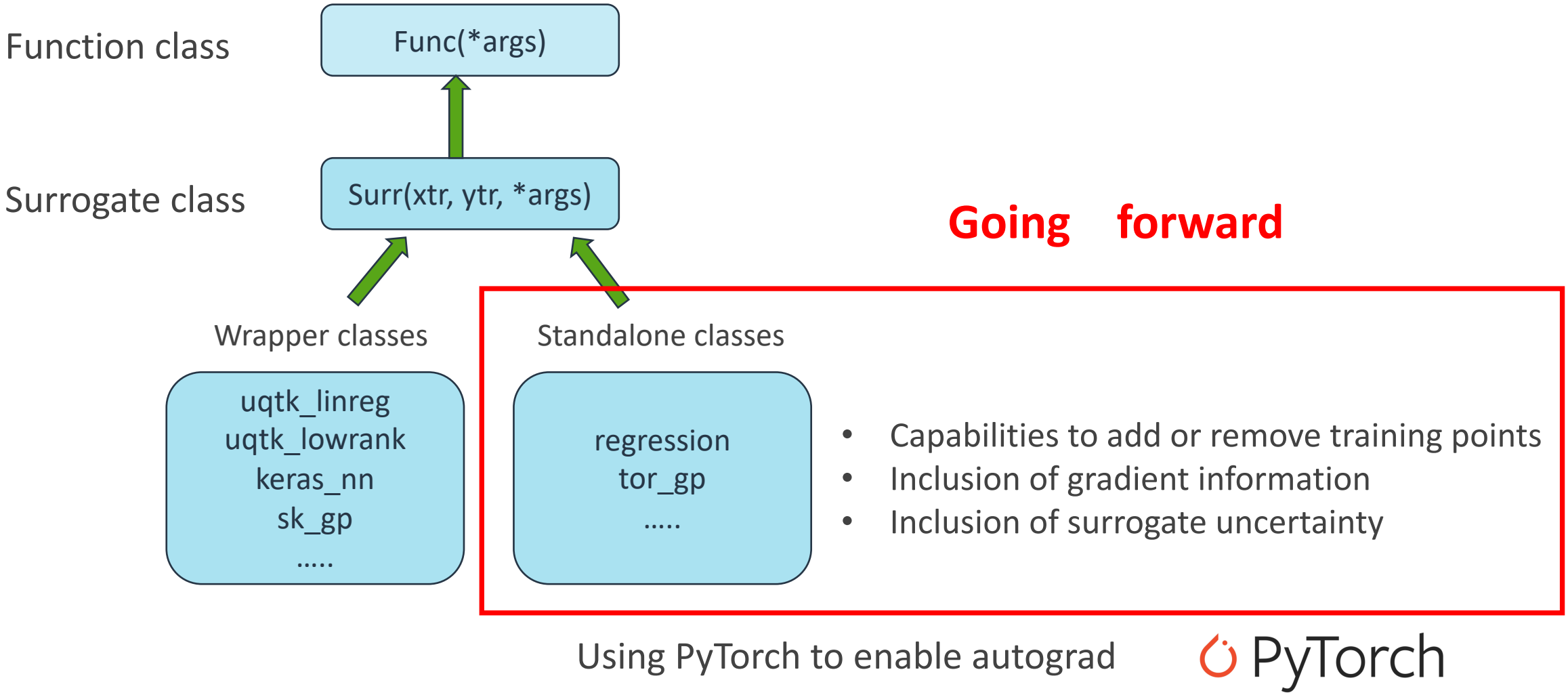
PyTorch

- Neural Networks (NN)

FitPy: code structure



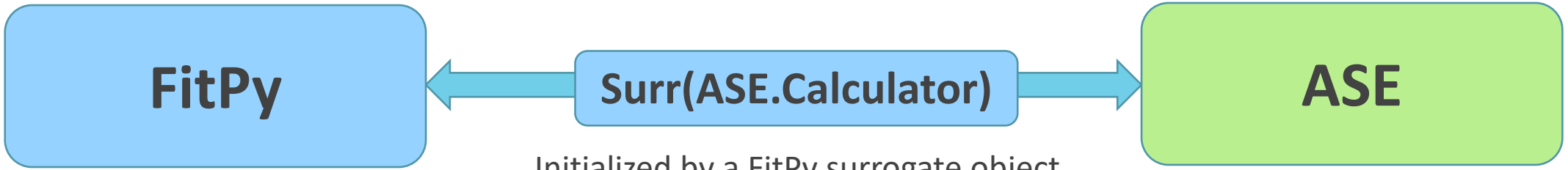
FitPy: code structure



Current capabilities

	Gradient information	Adaptive addition/removal of training points	Surrogate uncertainty
UQTk GP	No	No	Yes
UQTk PC/LSQ/BCS	No	No	Yes
UQTk Low-rank Tensors	No	No	No
Keras/Scikit/PyTorch (NN wrappers)	No	No	No
FitPy LinReg	Coming	Maybe	Yes
FitPy GP	Yes	Yes	Yes
Amp (Atomistic ML Package)	Yes	No	No

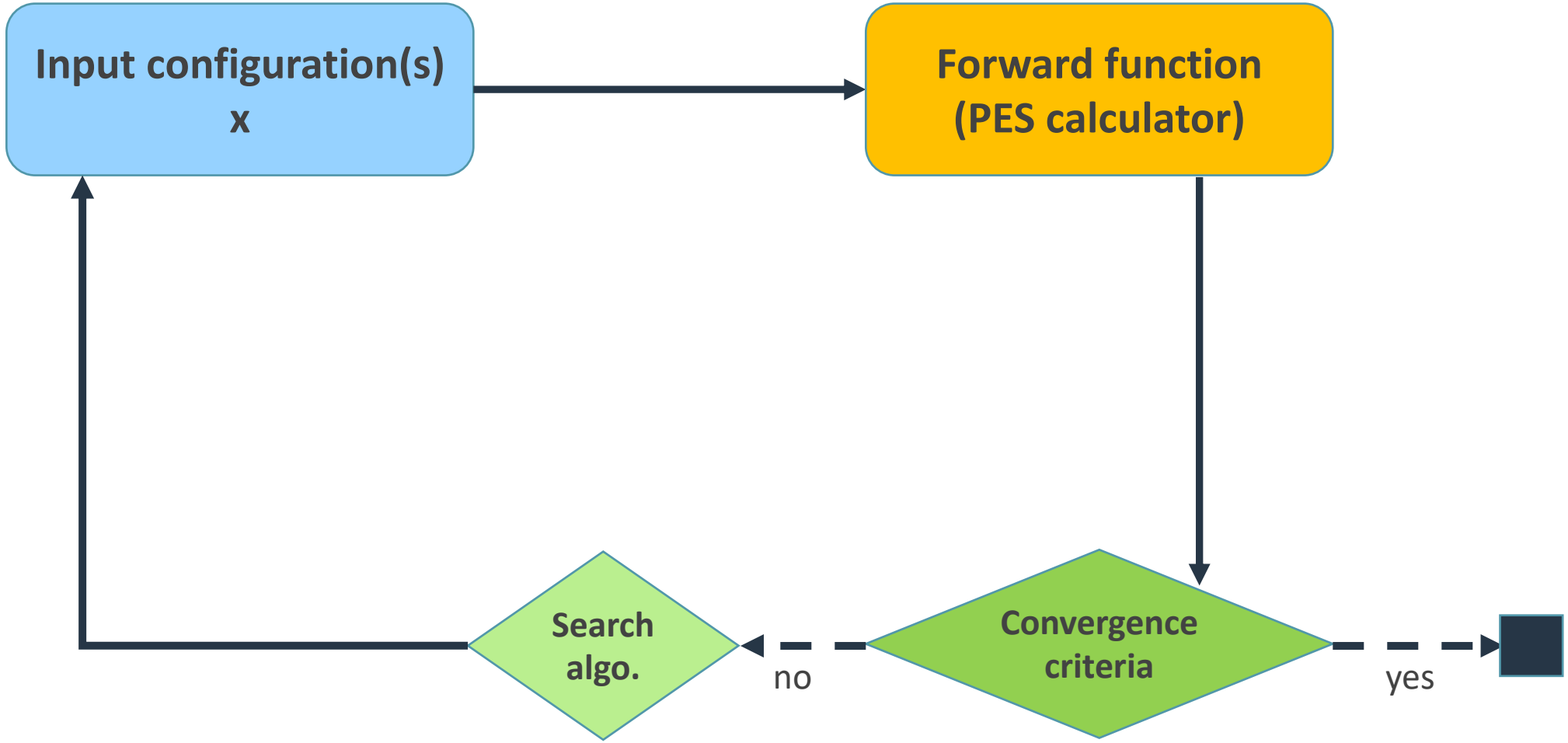
Connects to Atomic Simulation Environment via Surrogate Calculator



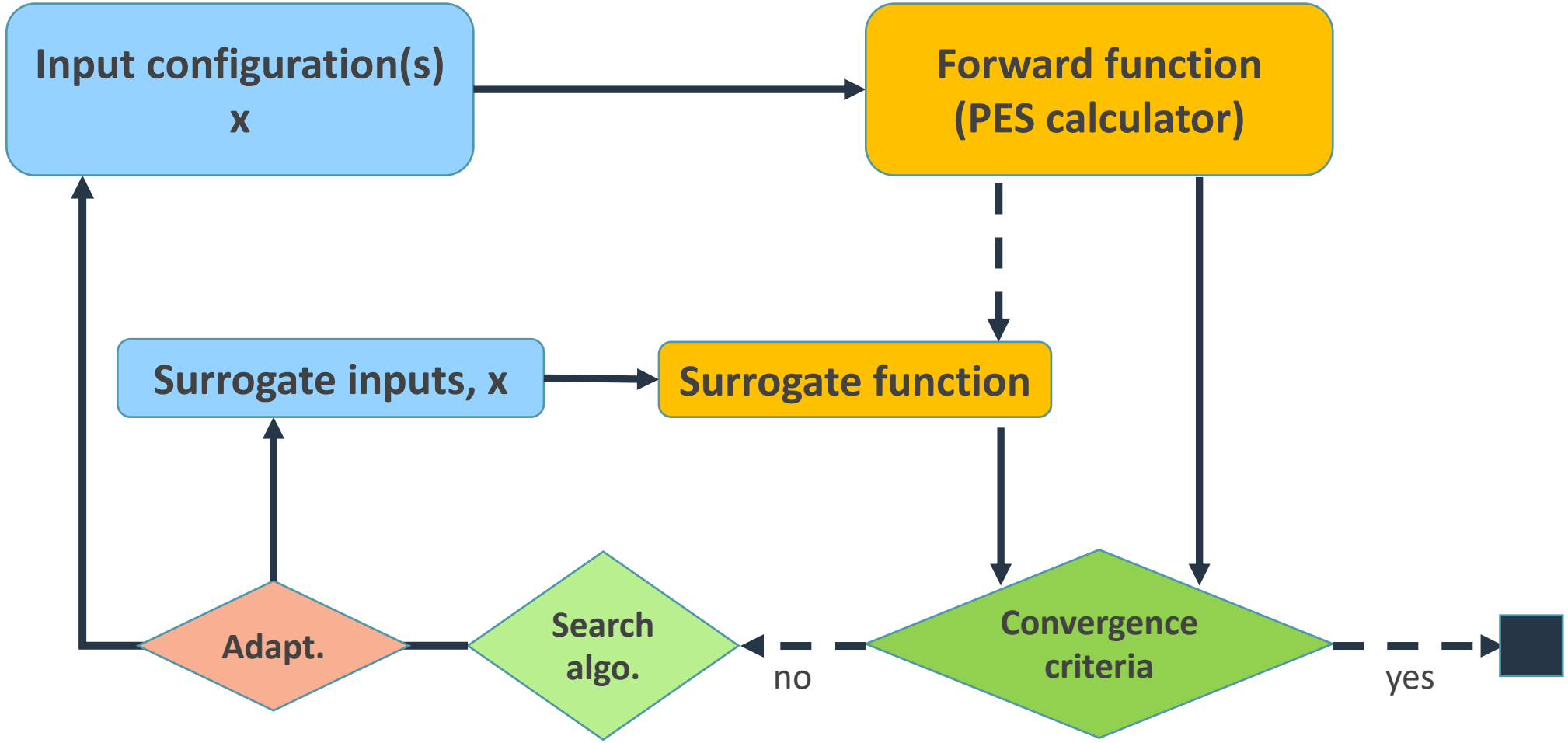
Initialized by a FitPy surrogate object
(which is derived FitPy function object)

Connecting with ASE will allow to tap into rich set of software and search algorithms

Typical Search Workflow



Typical Search Workflow



Focusing on two major surrogate types

- **Gaussian Process Regression (nonparametric):**
  - Implemented, cleanup pending
  - Because of linearity, gradient components are also Gaussian
  - PyTorch provides convenient machinery for automatic differentiation
    - E.g. *CatLearn* uses numerical finite difference derivative
  - Adaptive addition/removal of training points

- **Sparse Polynomial Regression (parametric):**
  - Implementation pending
  - Bayesian compressive sensing is necessary because of high-dimensionality
  - It is a GP afterall, but with a special (and degenerate) covariance.

Adaptivity is implemented via augmented training set

Training set…

- N input points: X is N x d
- N output values: Y is N x 1

Adaptivity is implemented via augmented training set

Training set…

- N input points: X is N x d
- N output values: Y is N x 1

… augmented with

- N grad. indicators: G is N x 1

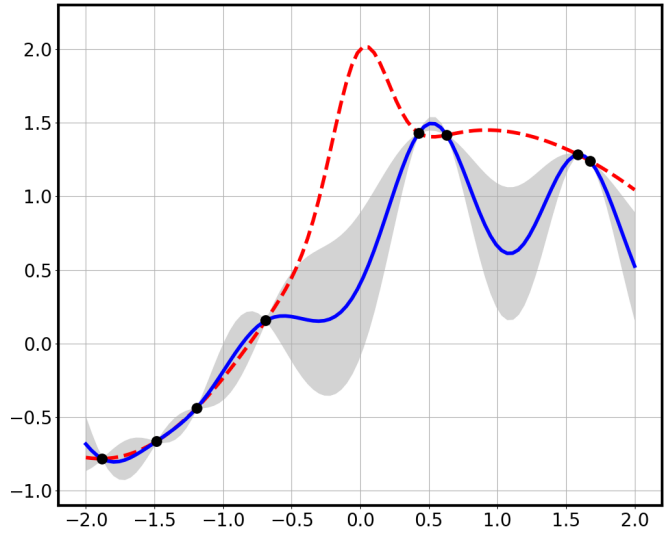
G(i)=0 means Y(i) is value (energy)
G(i)=k>0 means Y(i) is gradient wrt k-th dim. (force)

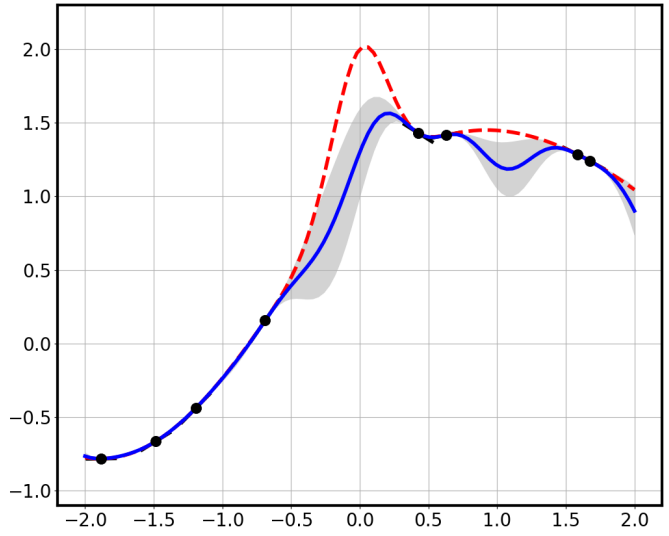
- Fitting and evaluation take G together with X and Y → Fit(X, Y, G), Eval(X, Y, G)

- Allows flexible addition of energy or force components

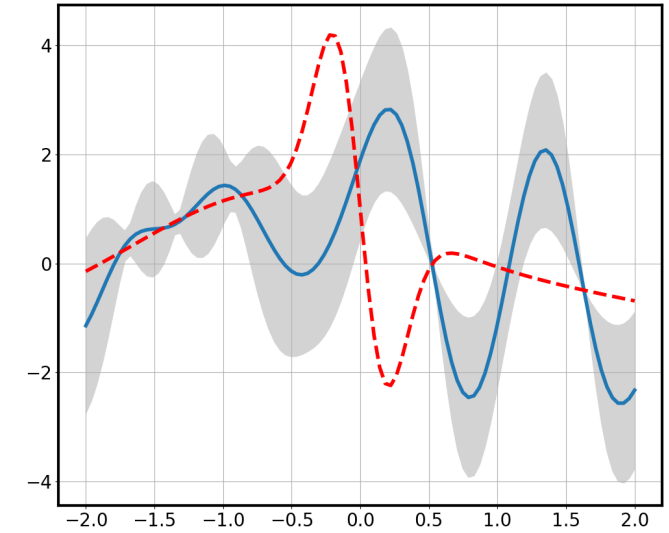
Gradient information helps

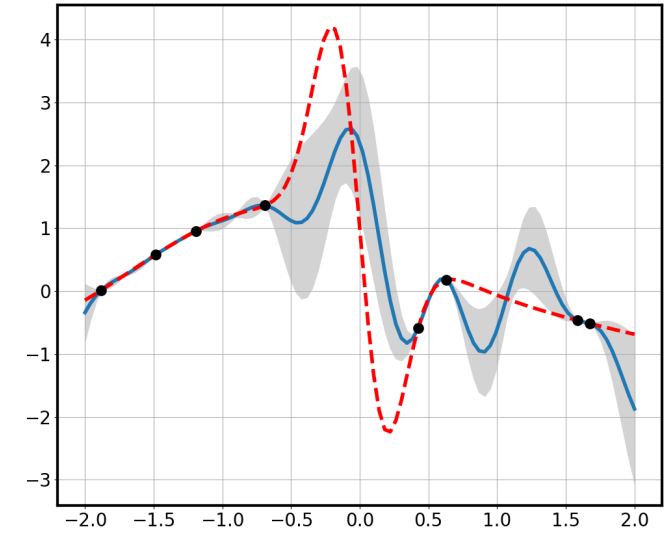
Function



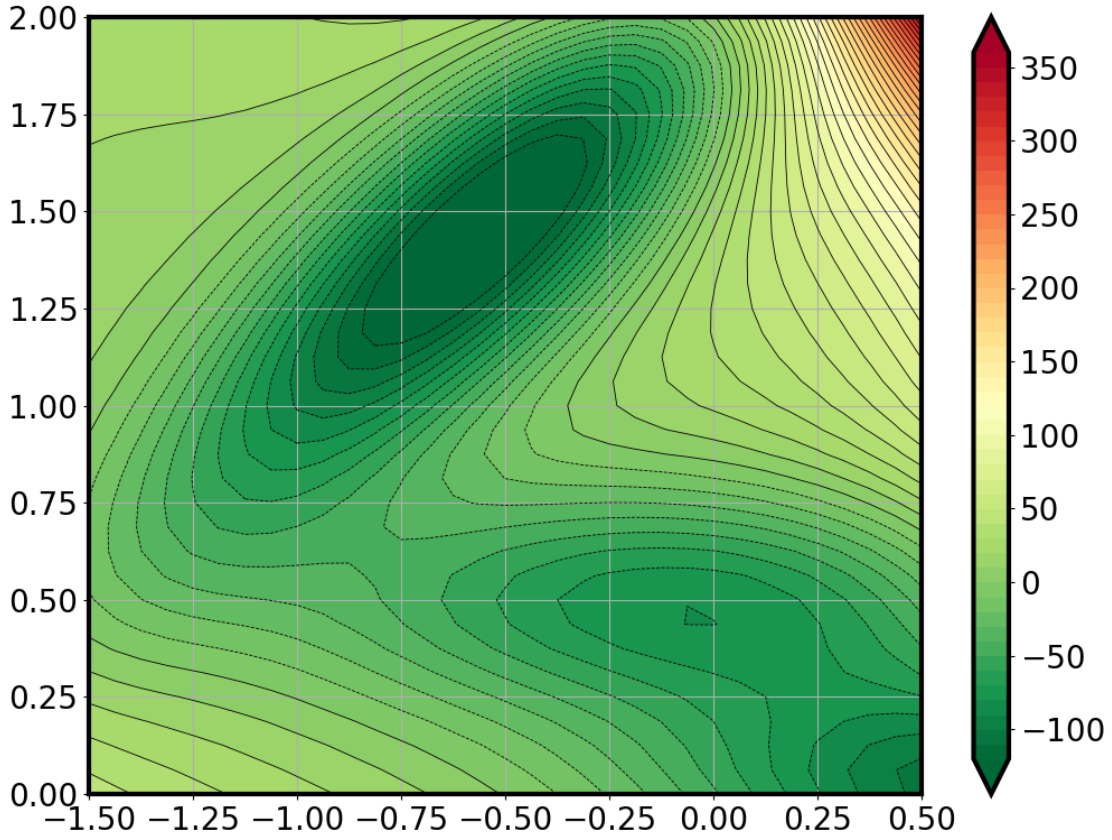


Derivative

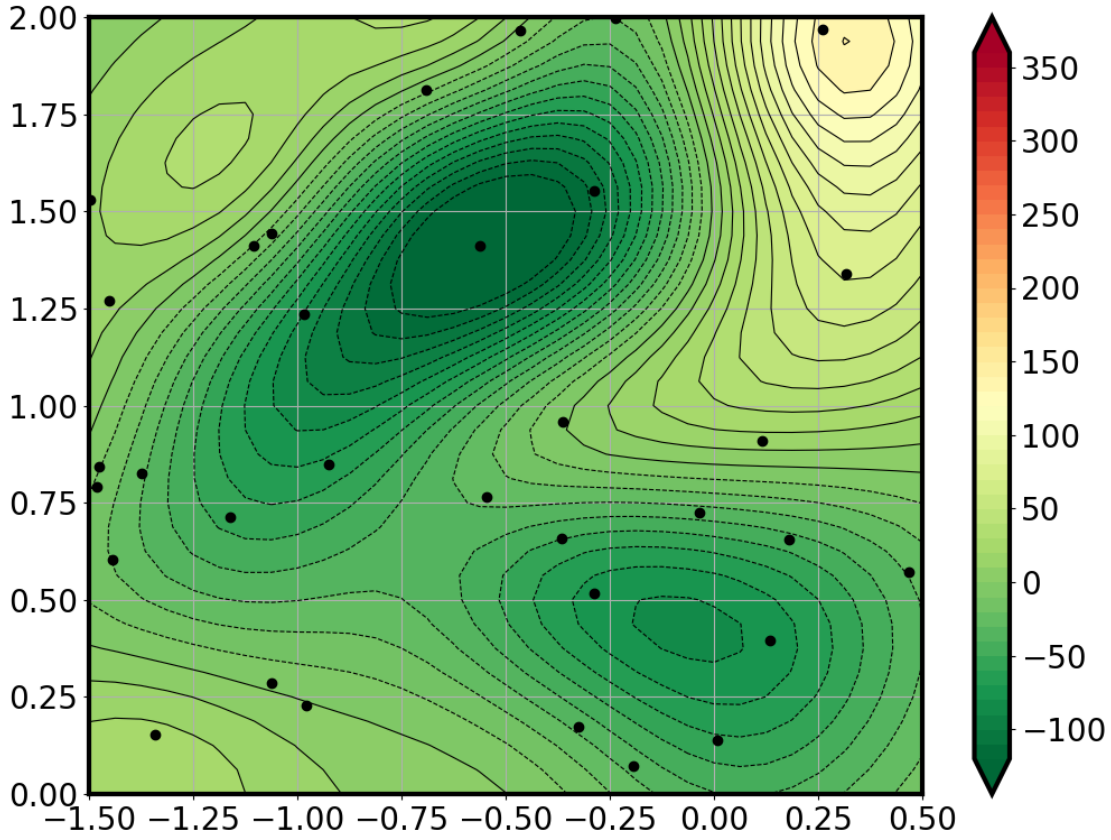




Gradient information helps

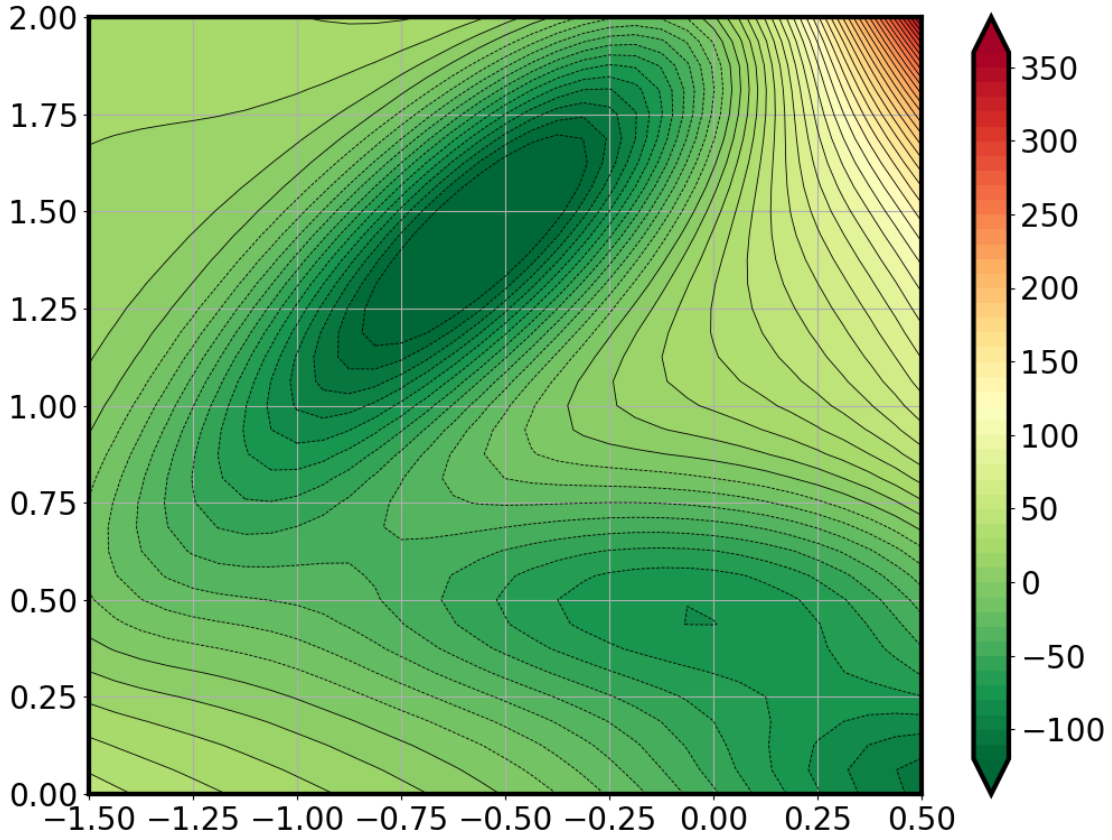


Muller-Brown PES

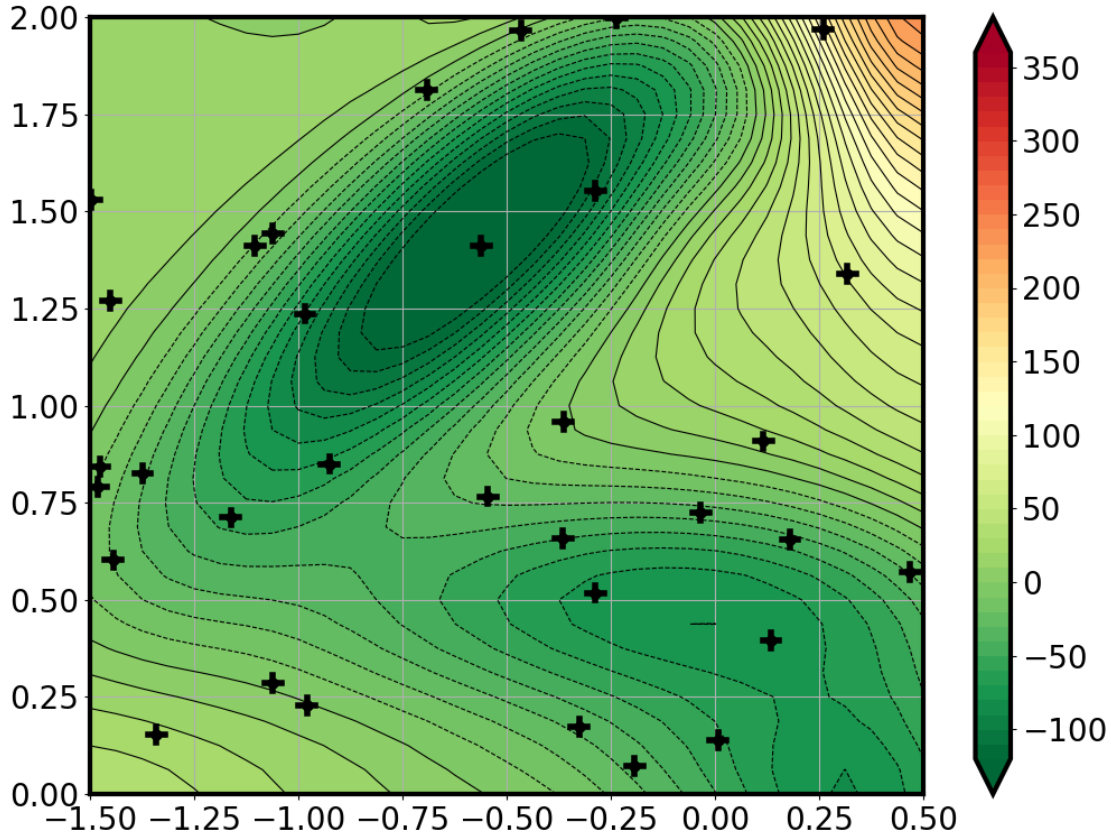


GP fit

Gradient information helps

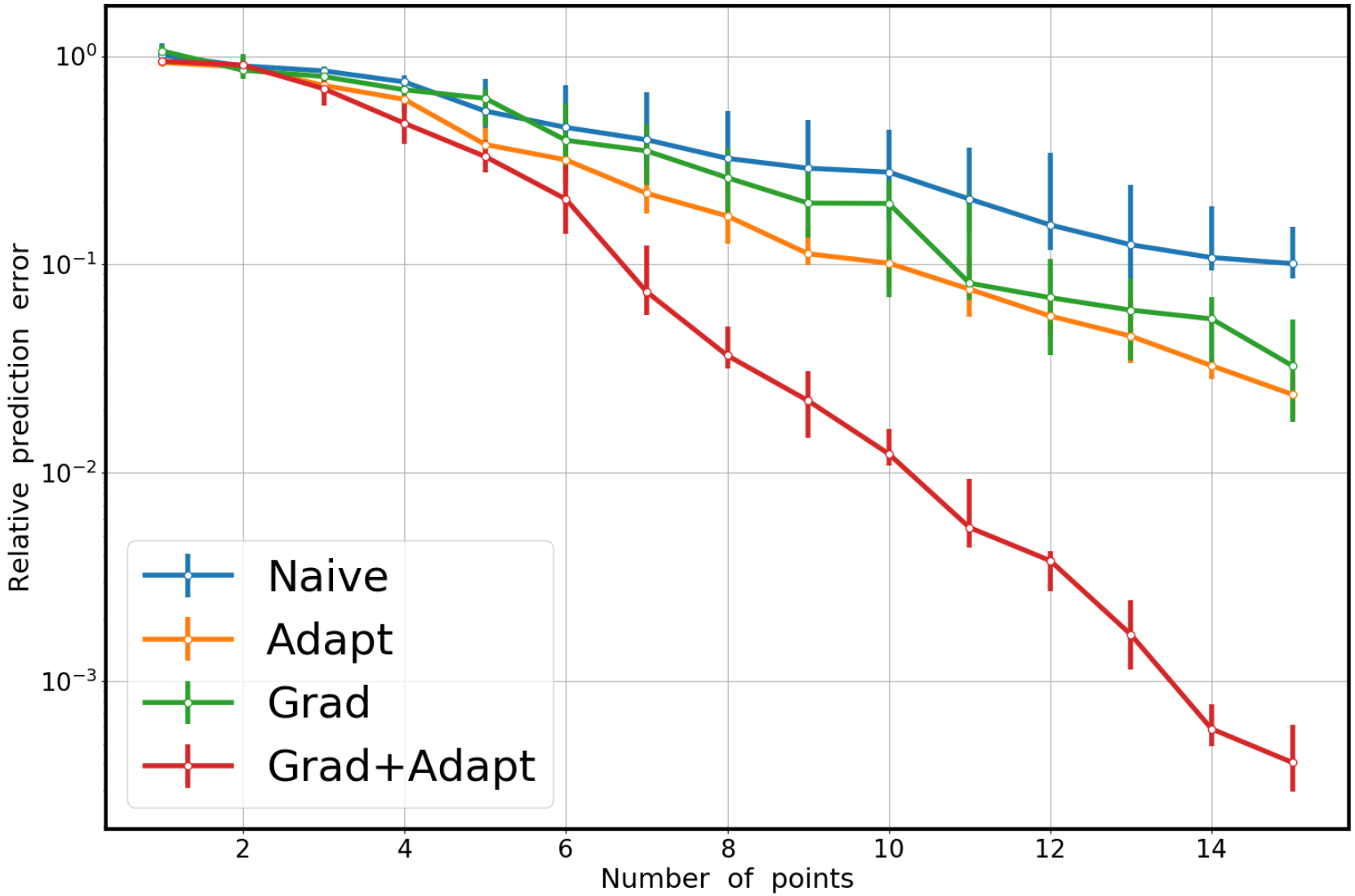


Muller-Brown PES

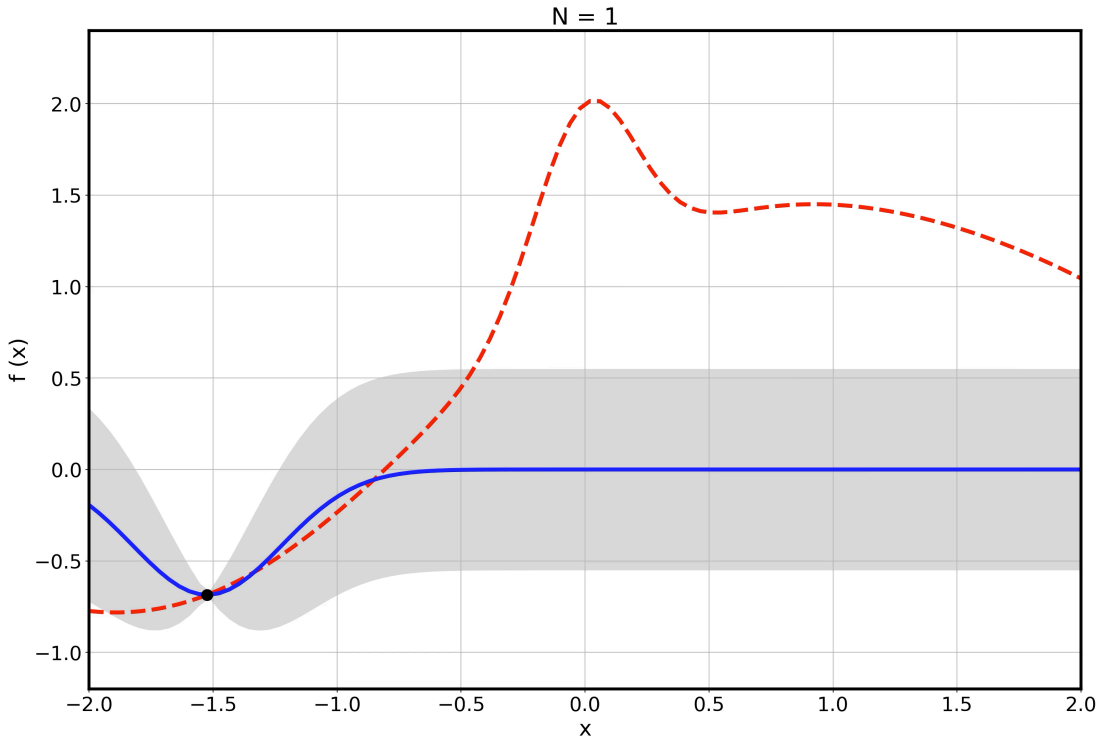


GP fit with gradients

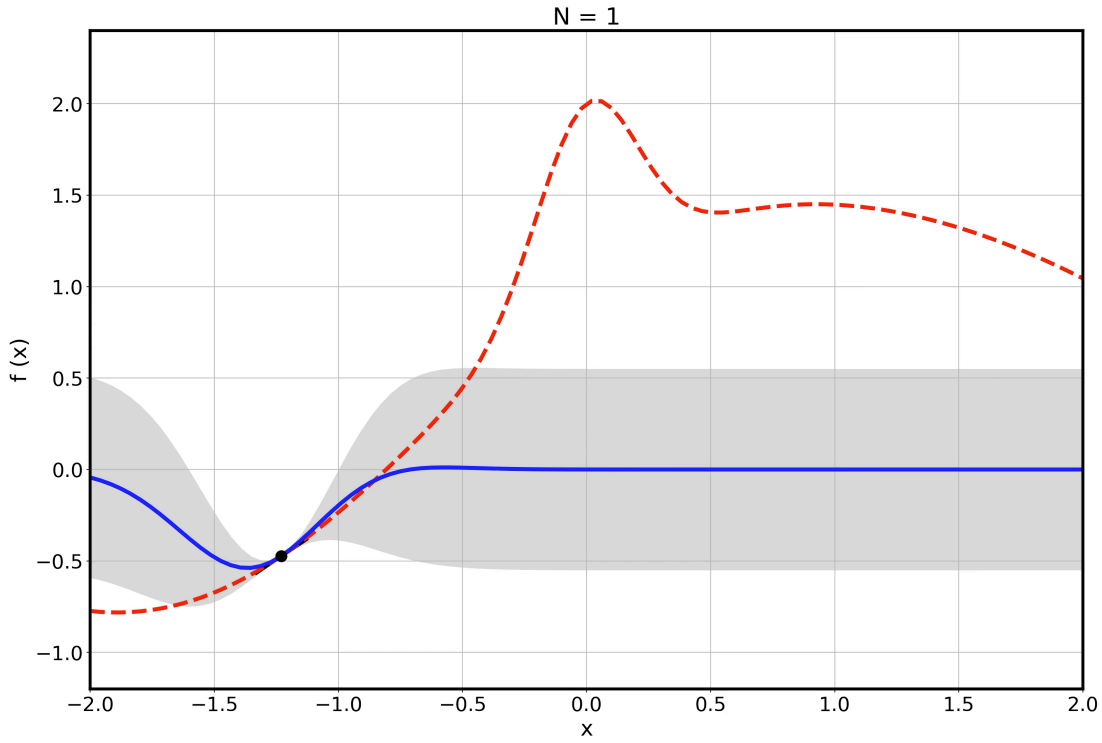
Adaptation helps even better



Adaptive point addition according to maximal variance in a randomly selected set



Naïve point addition

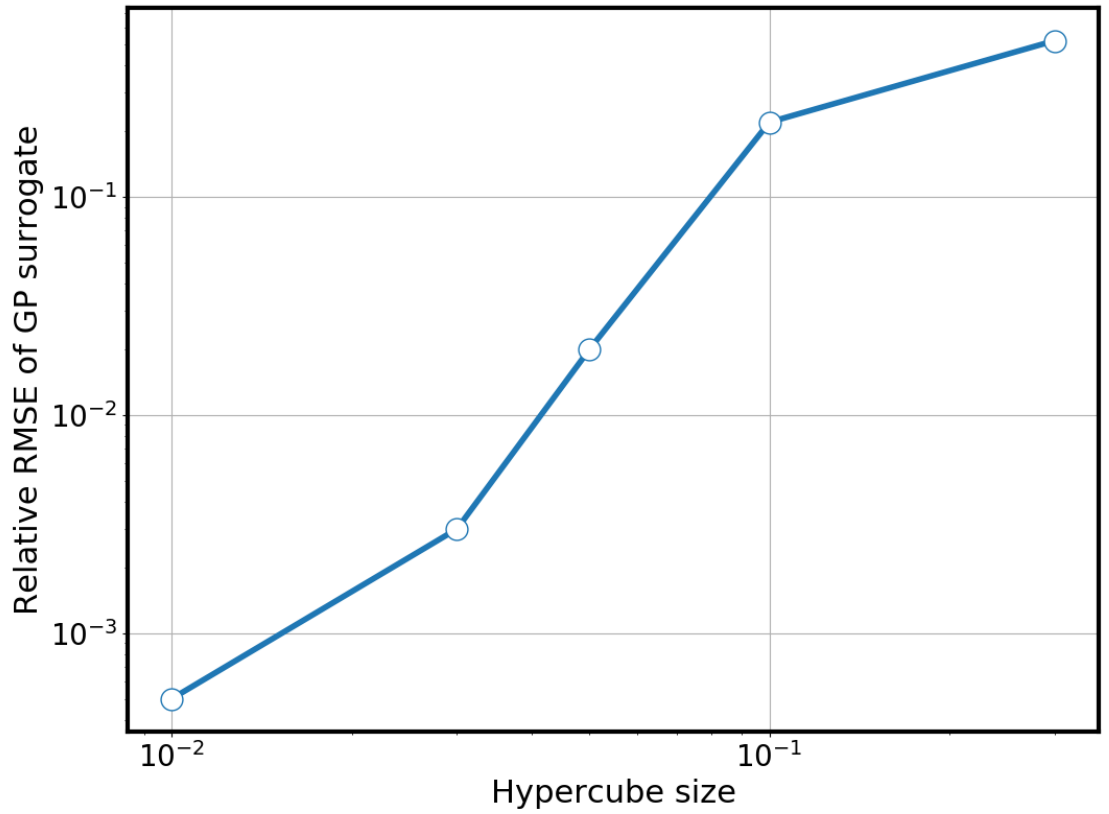


Adaptive point addition + derivatives

Locality matters

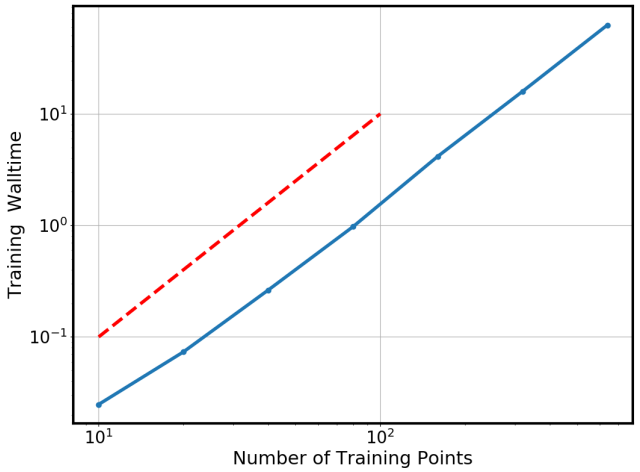
- O3 from PotLib, https://comp.chem.umn.edu/potlib/
- Dimensionality d=9
- Training size N=50
- The smaller the region of interest, the more accurate GP surrogate is.

No surprise, of course!

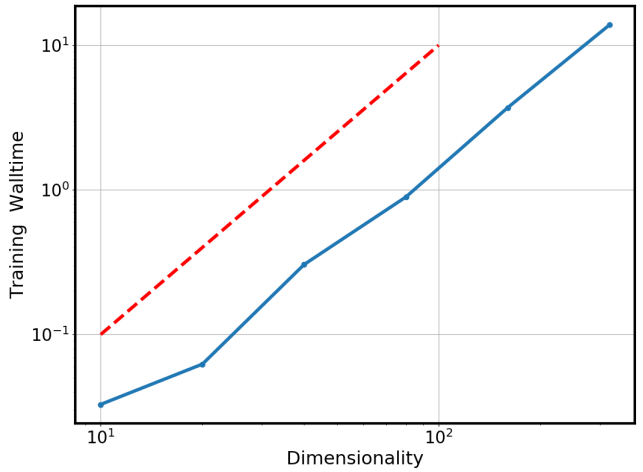


Scaling study: $T = O(N^2d^2)$

$T=O(N^2)$

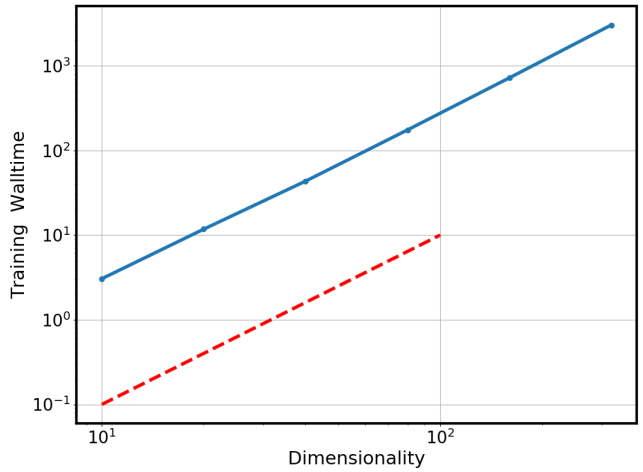


$T=O(d^2)$



Without gradients

$T=O(d^2)$



With gradients

Current

- Inclusion of **gradients**/forces
  - both for training and for evaluation
- Evaluation of predictive **uncertainties**
  - necessary for confidence assessment and adaptive methods
- **Adaptive** construction, locality, trust region
  - to support Sella or other search algorithms

Ongoing improvements

- Hyperparameter optimization,

as well as Max. Likelihood

- More kernels, not just square exponential
- Sparse polynomial regression via

Bayesian compressive sensing

- Usage of descriptors other than Cartesian

Plan ahead

- Planning a paper in Virtual Special Issue (VSI) of The Journal of Physical Chemistry A/B/C titled "Machine Learning in Physical Chemistry."

- Current target ML-NEB that scales:

including active learning ideas of locally adding and forgetting training points building on Koistinen et. al. work, as well as CatLearn