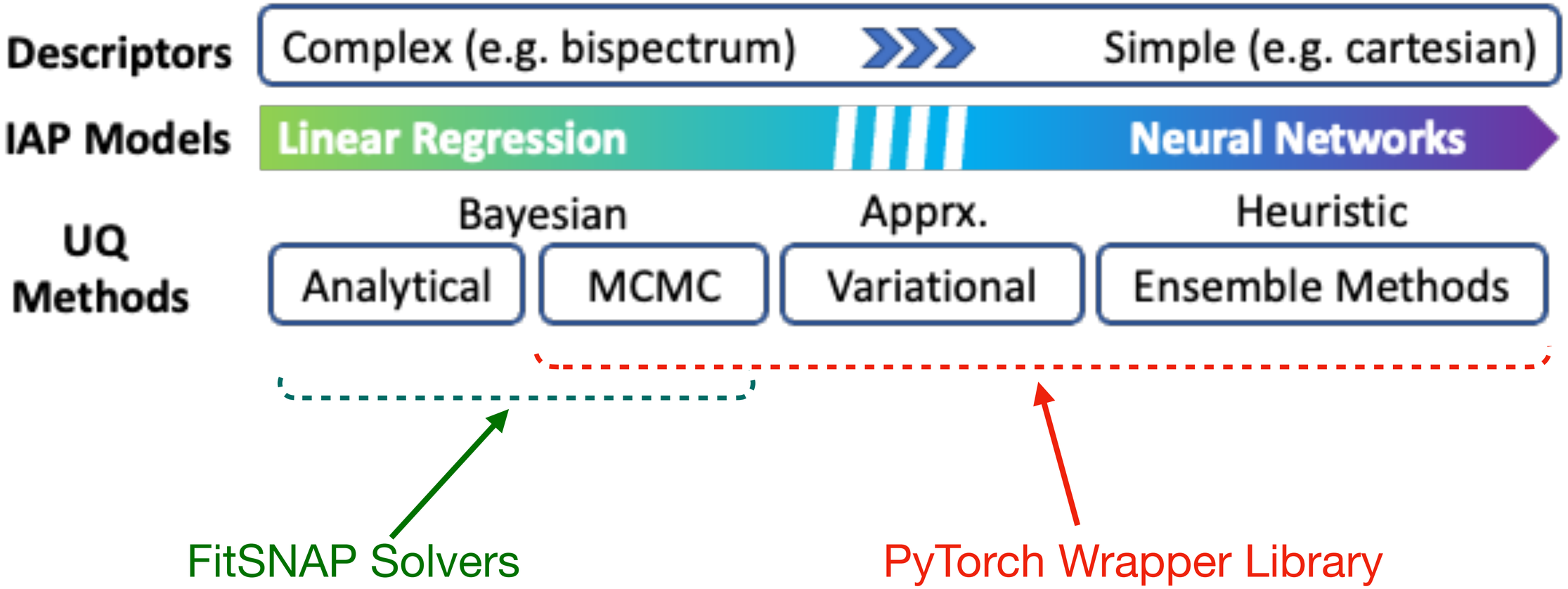
Model Error Estimation in Linear Regression

FusMatML Update

Jan 13, 2022

Khachik Sargsyan, Habib Najm (SNL-CA)

Equipping parametric fits with uncertainties



FitSNAP solvers with Uncertainty

Forked and created a UQ branch

uq ▾ FitSNAP / fitsnap3 / solvers /

This branch is 8 commits ahead of FitSNAP:master.

ksargsyan bug fix in MCMC ...

..

- anl.py
- bcs.py
- mcmc.py
- opt.py
- scalapack.py
- solver.py
- solver_factory.py
- svd.py
- template_solver.py
- tensorflowsvd.py
- merr.py

Analytical Bayesian linear regression

```
[SOLVER]
solver = ANL
nsam = 133
cov_nugget = 1.e-10
```

Bayesian compressive sensing (TBD, need bispectrum pruning)

```
[SOLVER]
solver = BCS
nsam = 133
```

MCMC

```
[SOLVER]
solver = MCMC
nsam = 133
mcmc_num = 1000
mcmc_gamma = 0.01
```

Optimization via scipy.optimize

```
[SOLVER]
solver = OPT
```

Model error, TBD

this talk

UQ solvers creates the requested number (nsam) of snapcoeff files, e.g.

Sample # 25

WZrC_pot_025.snapcoeff

```
# fitsnap fit generated on 2021-10-30 00:46:00.217256

3 31
W 0.4033335777 1.0
 -15.3500742786061313          #  B[0]
 0.0325450949477414861         #  B[1, 0, 0, 0]
 0.0786028816107458284         #  B[2, 1, 0, 1]
 -0.144382530698556222         #  B[3, 1, 1, 2]
 -0.738222137312165794         #  B[4, 2, 0, 2]
 -0.140285804611108733         #  B[5, 2, 1, 3]
 0.288874324985492814          #  B[6, 2, 2, 2]
 -0.0217902885756043087        #  B[7, 2, 2, 4]
 -0.377006851861265813         #  B[8, 3, 0, 3]
 -0.845242239468904089         #  B[9, 3, 1, 4]
 1.11733162664206276           #  B[10, 3, 2, 3]
 -0.053260813672681858?        #  B[11, 3, 2, 5]
```

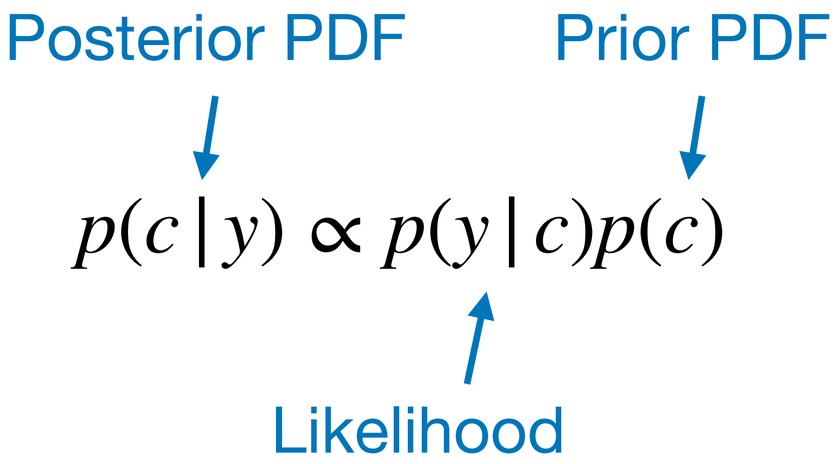
(Bayesian) Parameter Inference

- Given a model $f(x, c)$ and data $y_i = y(x_i)$, calibrate parameters c.
  - Linear model $y \approx Ac$ with coefficients c
  - NN model $y \approx NN_c(x)$ with weights/biases c

- Weighted least-squares fit: $$c^* = \text{argmin}_c \sum_{i=1}^{N} w_i^2 \left(f(x_i, c) - y_i\right)^2$$

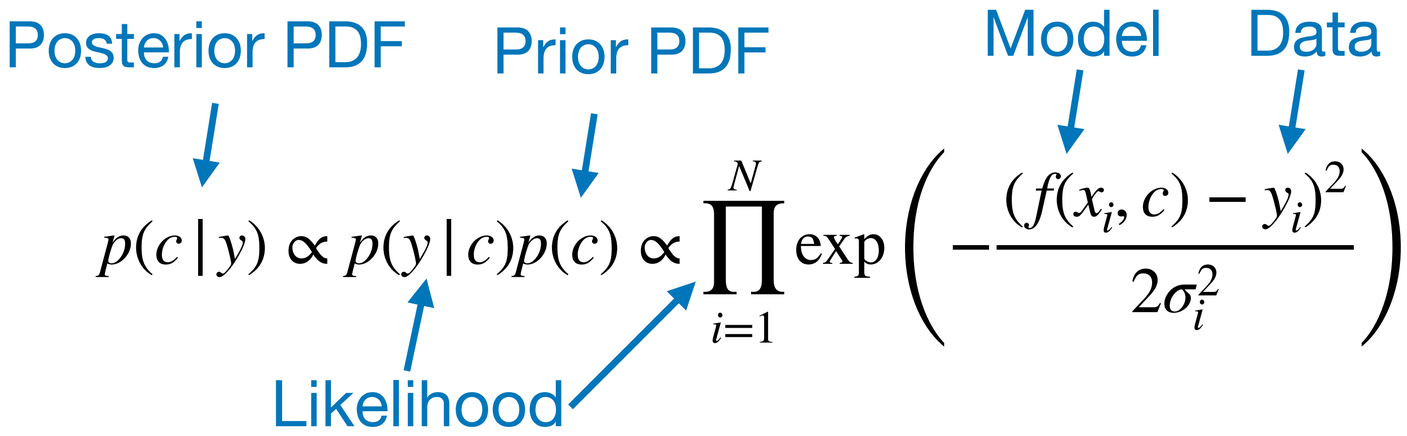
- Bayesian equivalent: $$p(c\,|\,y) \propto p(y\,|\,c)p(c) \propto \prod_{i=1}^{N} \exp\left(-\frac{(f(x_i, c) - y_i)^2}{2\sigma_i^2}\right)$$

Posterior PDF sampling via MCMC



- ✦ The likelihood requires assumptions regarding model/data relationships
- ✦ No closed form expression for posterior PDF unless very specialized likelihoods are used
- ✦ Need to resort to sampling the posterior, rather than evaluating directly
- ✦ Markov chain Monte Carlo is the main vehicle for posterior sampling

Crucial piece: assumptions for likelihood, or data model, or noise model



- ✦ Prior contains previous knowledge or regularization
- ✦ Likelihood contains data noise modeling assumptions,

e.g. $y_i = f(x_i, c) + \sigma_i \epsilon_i$, where $\epsilon_i \sim \mathcal{N}(0,1)$

Data Model

Elephant in the room: model is assumed to be *the* correct model behind data

Model

Data err.

$$y_i = f(x_i, c) + \sigma_i \epsilon_i$$

Truth

Model ≠ Truth

Ignoring model error hurts in a few ways:

✦ One gets biased estimates of parameters c (crucial if the model is physical, and/or c is propagated through other models)

✦ More data leads to overconfident predictions (we become more and more certain about the wrong values of the data)

✦ More evident when there is no (observational/experimental) data error: *e.g.* DFT is data, and IAP is model

Posterior uncertainty does not capture true discrepancy

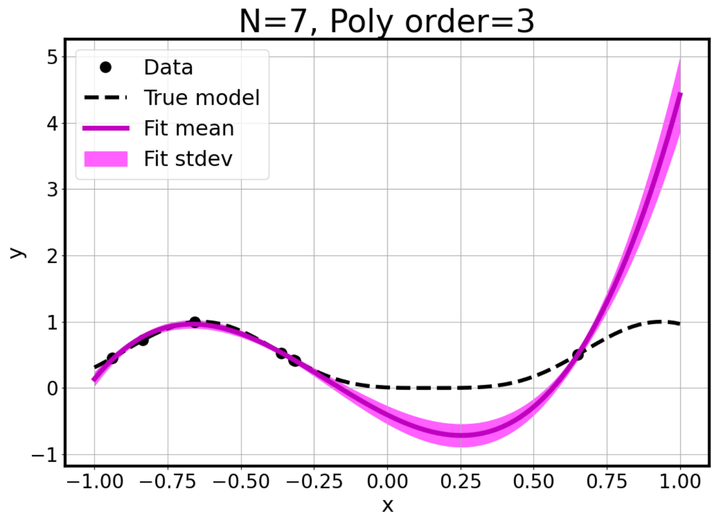
Synthetic data

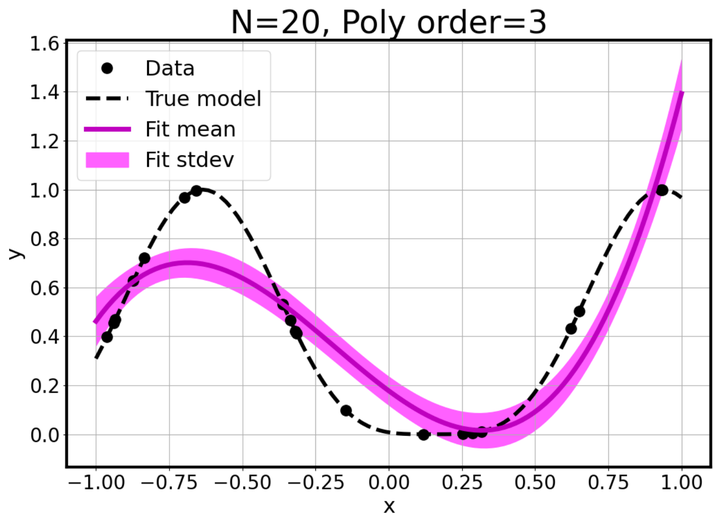
$$y(x) = \sin^4(2x - 0.3)$$

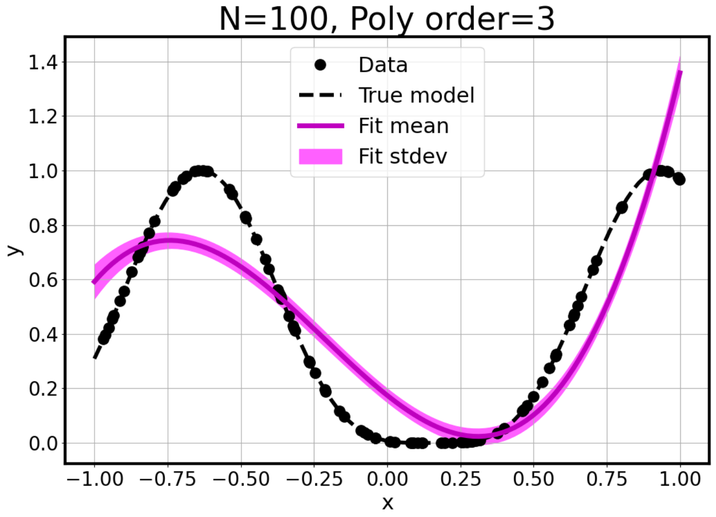
Cubic fit

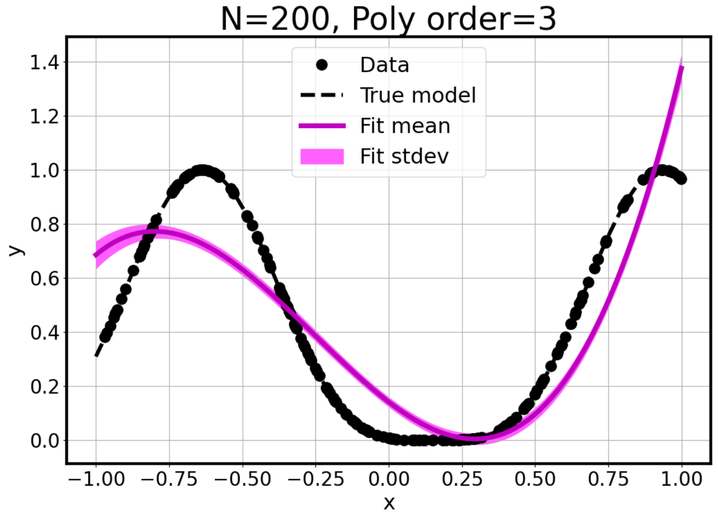
$$y_i \approx \sum_{k=0}^{3} c_k B_k(x)$$

More data leads to overconfident prediction









Capturing Model Error in Likelihood (a.k.a. Data Model)

External correction (Kennedy-O'Hagan):

$$y_i = f(x_i, c) + \delta(x_i) + \sigma_i \epsilon_i$$

- Kennedy, O'Hagan, "Bayesian Calibration of Computer Models". *J Royal Stat Soc: Series B (Stat Meth),* 63: 425-464, 2001.

Internal correction (embedded model error):

$$y_i = f(x_i, c + \delta(x_i)) + \sigma_i \epsilon_i$$

- Allows meaningful usage of calibrated model
- 'Leftover' noise term even with no data error
- Respects physics (not too relevant in our context)

- Sargsyan, Najm, Ghanem, "On the Statistical Calibration of Physical Models". *Int. J. Chem. Kinet.*, 47: 246-276, 2015.
- Sargsyan, Huan, Najm, "Embedded Model Error Representation for Bayesian Model Calibration". *Int. J. Uncert. Quantif.*, 9(4): 365-394, 2019.

Embedded Model Error for Linear Regression Models

$$y_i \approx \sum_{k=0}^{P} c_k B_k(x) + \sigma_i \epsilon_i$$

Note:

No formal distinction between internal and external corrections, but internal allows for interpretation and model-informed error

'Embed' uncertainty in all (or selected) coefficients

Model Model error

$$y_i \approx \sum_{k=0}^{P} (c_k + d_k \xi_k) B_k(x) = \sum_{k=0}^{P} c_k B_k(x) + \sum_{k=0}^{P} d_k B_k(x) \xi_k$$

(still Gaussian, but correlated, and model-informed)

Embedded Model Error: likelihood options

Classical data model

$$y_i \approx \sum_{k=0}^{P} c_k B_k(x) + \sigma_i \epsilon_i$$

$$p(c\,|\,y) \propto \prod_{i=1}^{N} \exp\left(-\frac{(\sum_{k=0}^{P} c_k B_k(x_i) - y_i)^2}{2\sigma_i^2} \right)$$

MCMC sampling of c

Embedded model error

$$y_i \approx \sum_{k=0}^{P} (c_k + d_k \xi_k)\, B_k(x) \quad = \quad \sum_{k=0}^{P} c_k B_k(x) + \sum_{k=0}^{P} d_k B_k(x) \xi_k$$

Option 1 (IID)

$$p(c, d\,|\,y) \propto \prod_{i=1}^{N} \exp\left(-\frac{(\sum_{k=0}^{P} c_k B_k(x_i) - y_i)^2}{2\sum_{k=0}^{K} d_k^2 B_k(x_i)^2} \right)$$

MCMC sampling of c, d

or

simply optimize the posterior for c, d

Embedded Model Error: likelihood options

Classical data model

$$y_i \approx \sum_{k=0}^{P} c_k B_k(x) + \sigma_i \epsilon_i$$

$$p(c\,|\,y) \propto \prod_{i=1}^{N} \exp\left(-\frac{(\sum_{k=0}^{P} c_k B_k(x_i) - y_i)^2}{2\sigma_i^2} \right)$$

MCMC sampling of c

Embedded model error

$$y_i \approx \sum_{k=0}^{P} (c_k + d_k \xi_k)\, B_k(x) \quad = \quad \sum_{k=0}^{P} c_k B_k(x) + \sum_{k=0}^{P} d_k B_k(x) \xi_k$$

Option 2 (ABC)

$$p(c, d\,|\,y) \propto \prod_{i=1}^{N} \exp\left(-\frac{(\sum_{k=0}^{P} c_k B_k(x_i) - y_i)^2 + (\sqrt{\sum_{k=0}^{P} d_k^2 B_k^2(x_i)} - \alpha\,|\sum_{k=0}^{P} c_k B_k(x_i) - y_i\,|)^2}{2\epsilon^2} \right)$$

Pushed forward predictive uncertainty captures the true discrepancy from the data

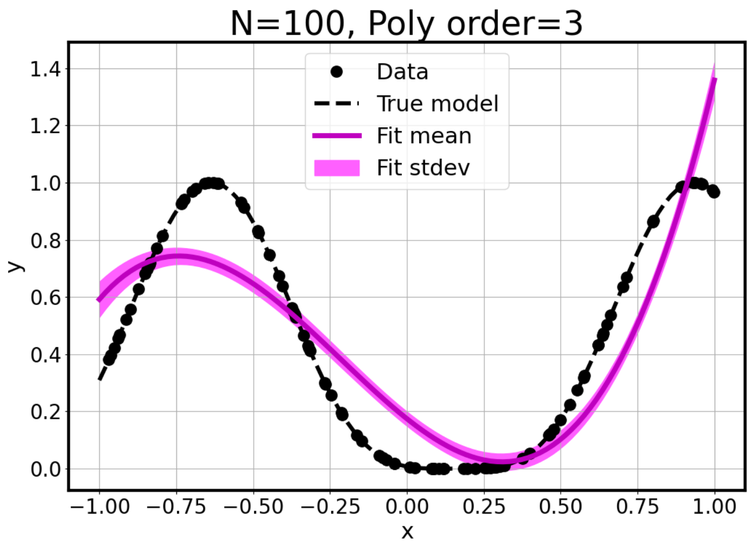
Synthetic data

$$y(x) = \sin^4(2x - 0.3)$$

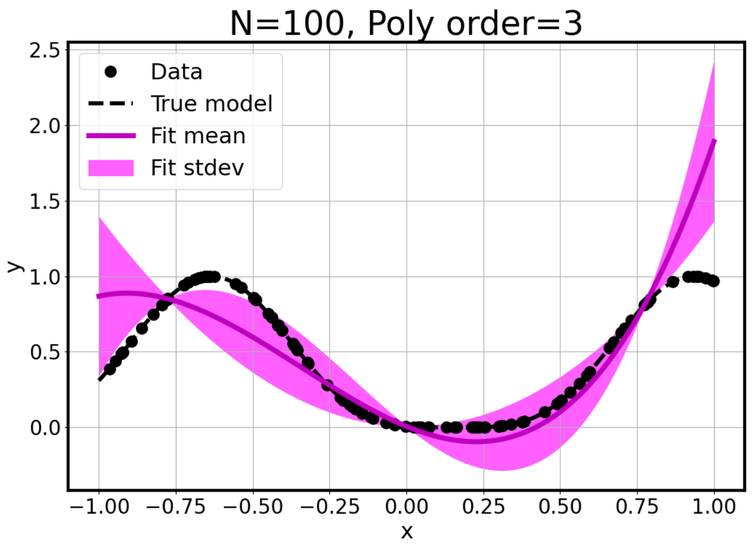
Cubic fit

$$y_i \approx \sum_{k=0}^{3} c_k B_k(x)$$

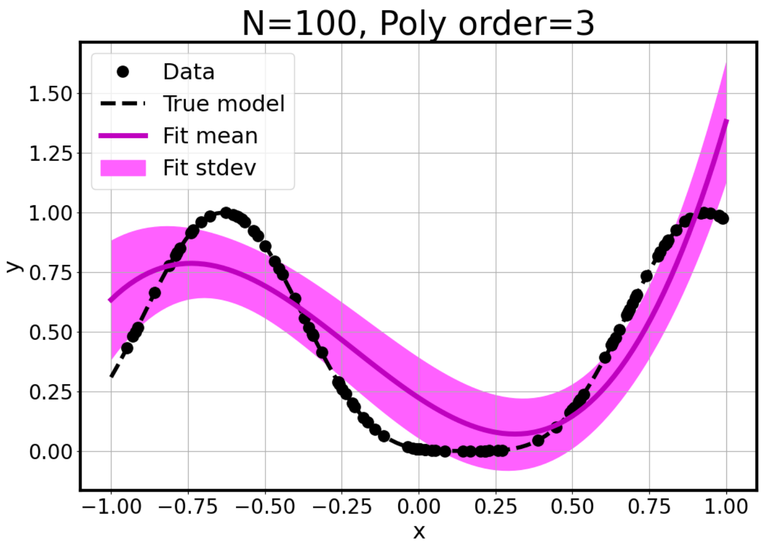
Classical case



Model error, IID likelihood



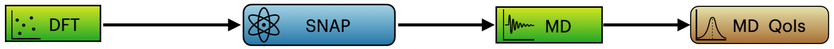
Model error, ABC likelihood



Two use cases to hone the methods

W-ZrC dataset

Uncertainty propagation through MD



with Ember Sikorski

Entropy dataset

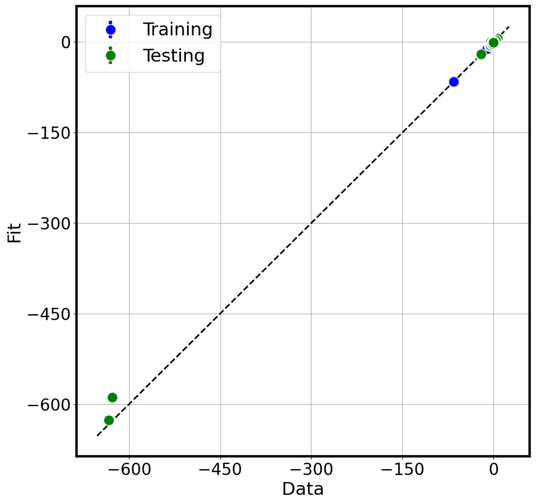
Validation and generalization of UQ-enabled SNAP potentials

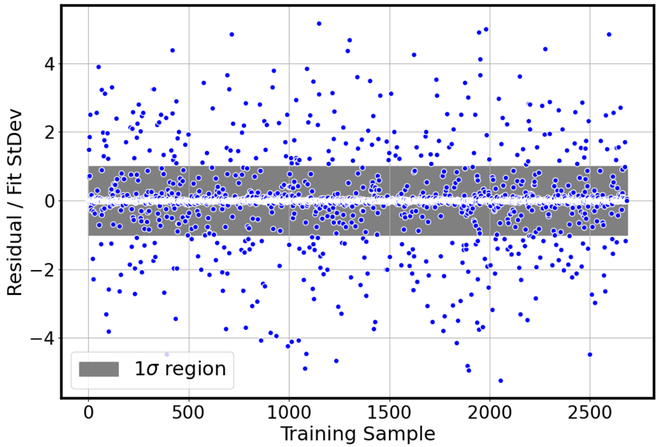
with David Montes de Oca Zapiain

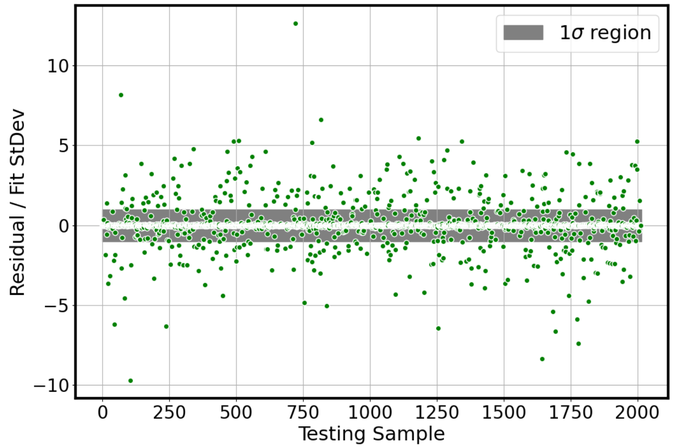
Regular meetings and Slack comm. with Ember, Megan, David and Mary Alice

W-ZrC Dataset

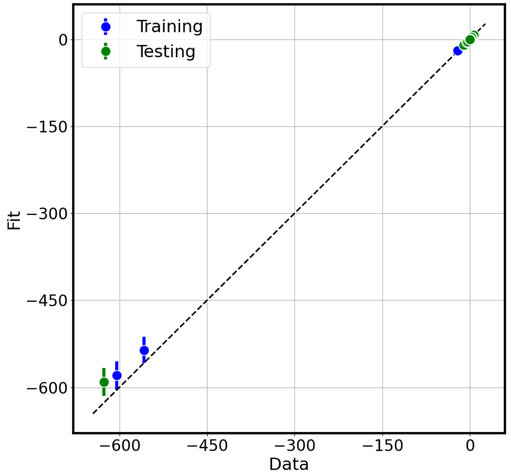
Uncertainty without model error

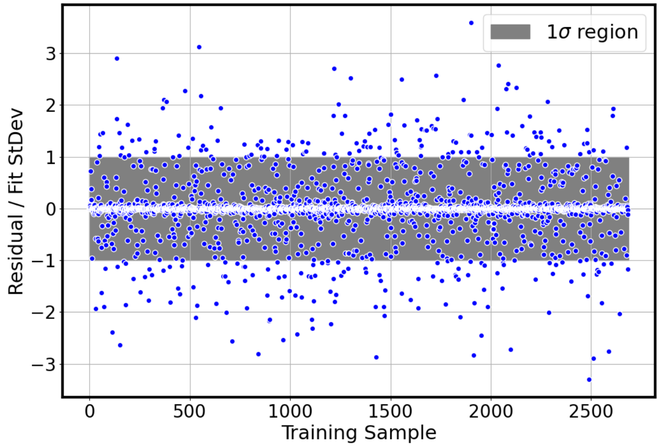


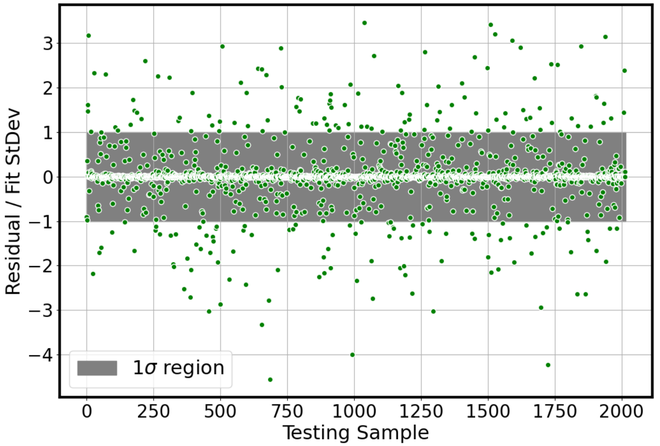




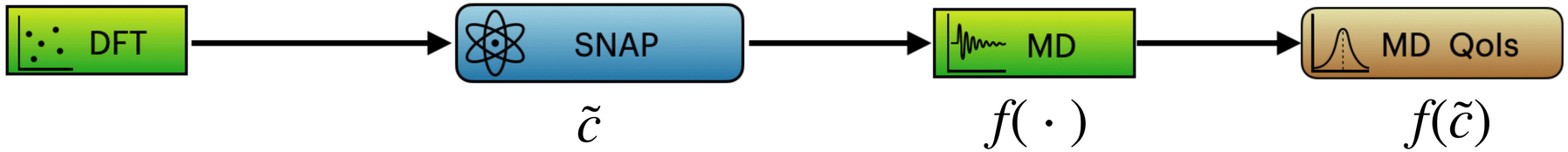
Uncertainty with model error







W-ZrC Dataset: next step



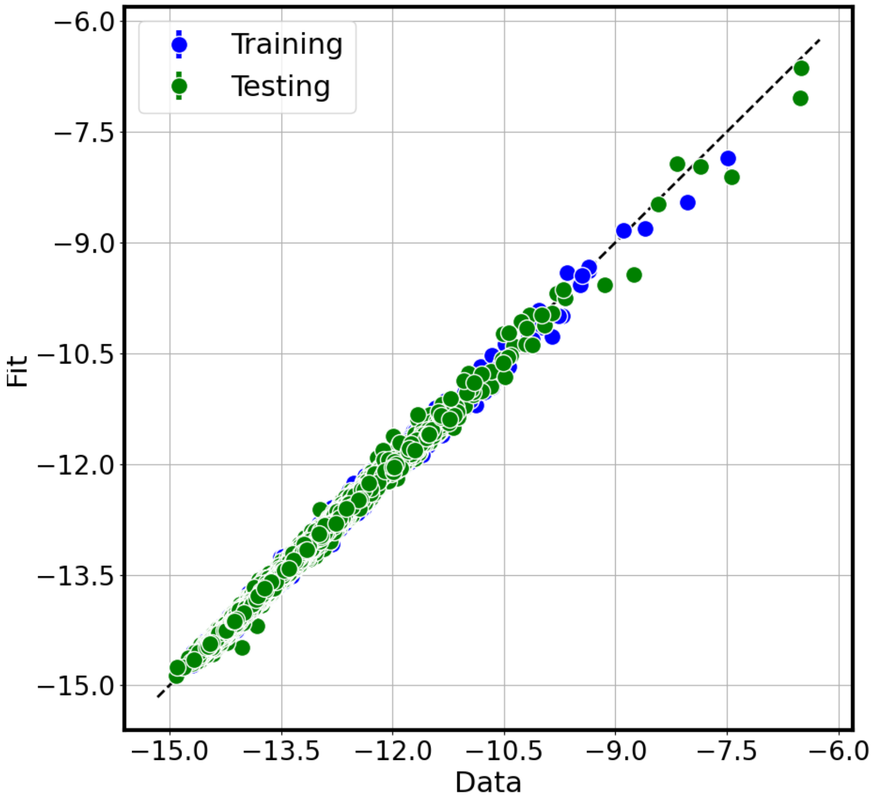
$$y_i \approx \sum_{k=0}^{P} \underbrace{(c_k + d_k \xi_k)}_{\tilde{c}} B_k(x)$$

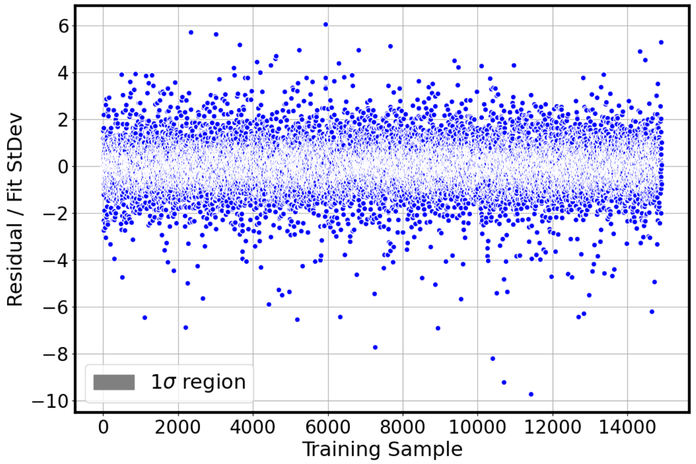
SNAP coefficients form a first order Gauss-Hermite Polynomial Chaos (PC)

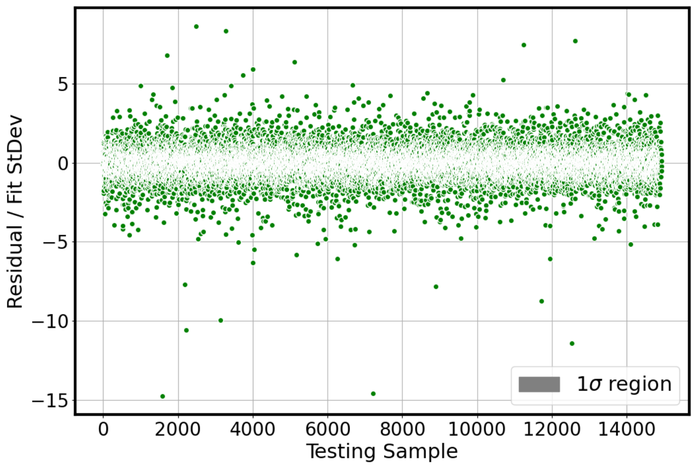
- Sample SNAP coefficients
- Evaluate MD QoIs
- Build PC expansion for MD QoIs
- Evaluate PDF/statistics of QoIs

Stay tuned for the next update

Entropy Dataset







Work in progress with David Montes de Oca Zapiain

Several challenges/choices

- Embedding type: *e.g.*

additive $y_i \approx \sum_{k=0}^{P} (c_k + d_k \xi_k)\, B_k(x)$ or multiplicative $y_i \approx \sum_{k=0}^{P} (c_k + c_k d_k \xi_k)\, B_k(x)$

- Degenerate (Gaussian) likelihoods: resort to approximate Bayesian computation (ABC) or independent (IID) assumptions

- Difficult posterior PDFs for MCMC, choice of priors for embedding parameters
- Which coefficients to embed the model error in? Several ideas, work in progress…
- Connect predictive uncertainty and the residual error with an extrapolation metric
- Major challenge: data sizes are large, linear algebra chokes

Summary

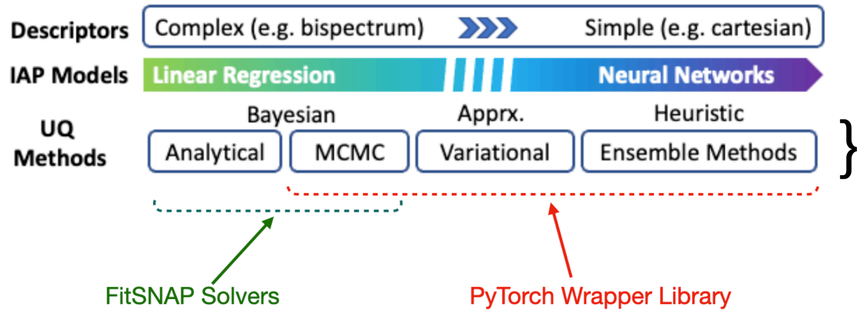
- Data model assumptions are crucial
- Embedded model error leads to data model with baked-in uncertainty
  - Meaningful model-error uncertainty capturing the true residual
  - Non-negligible coefficient uncertainty that can be propagated through MD
- Many modeling/method decisions to make
  - Hard to make it automated, but the plan is to expose them to user input file

---

- Logan Williams - starts January 24, 2022 as a postdoc
- We are also looking for an intern to help while searching for the second postdoc

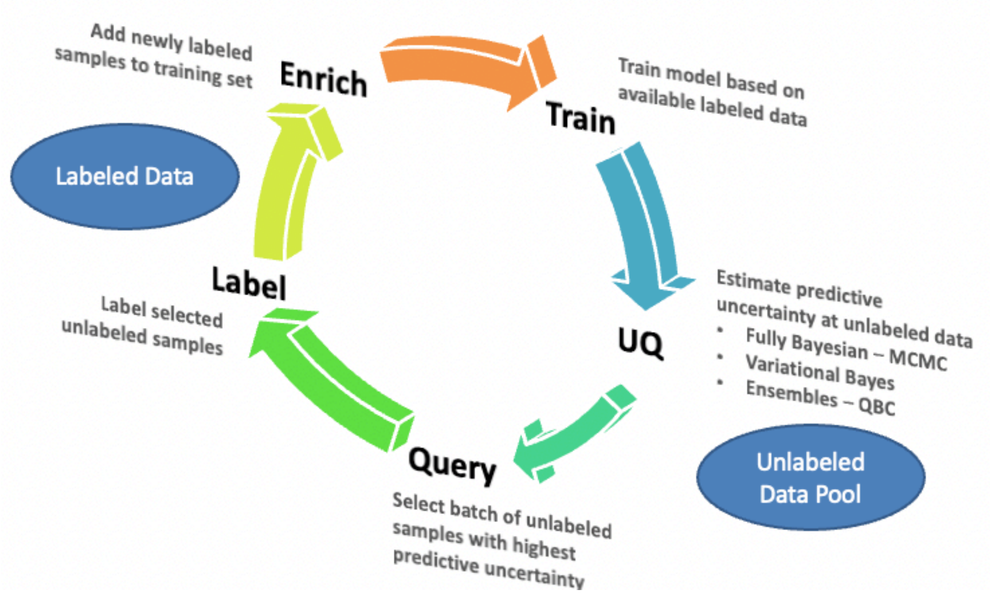
Extras

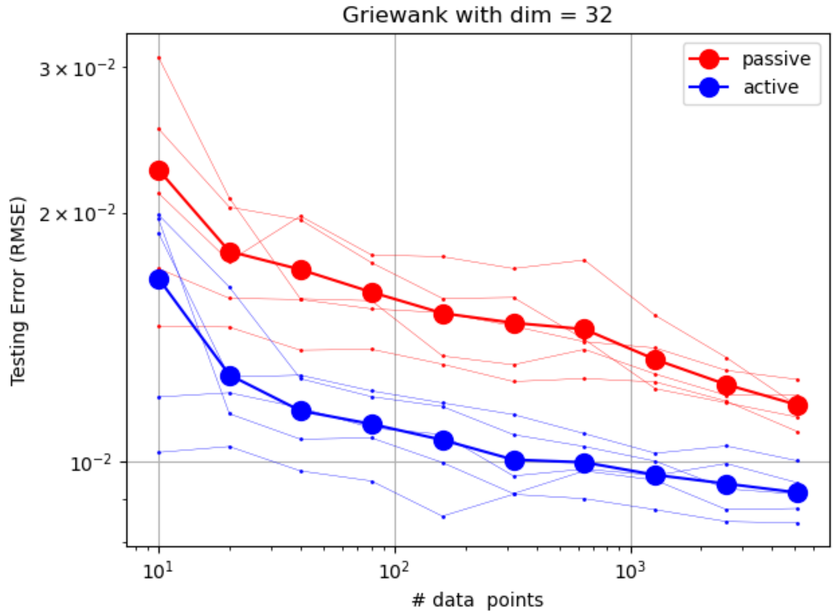
Active Learning: preliminaries



Active Learning

- Query-by-Committee is the method to go for now
- Promising results on synthetic models
- Exploring / gaining intuition on when AL works well





Plotting option added for quick visuals

```
[EXTRAS]
plot = 1
```

Options are 0, 1, 2.

Plots 'diagonal' plots (DFT-vs-SNAP) for all groups, weighted and unweighted,

Requires matplotlib, and may take time to generate all png files.

Option 2 plots with errorbars.