

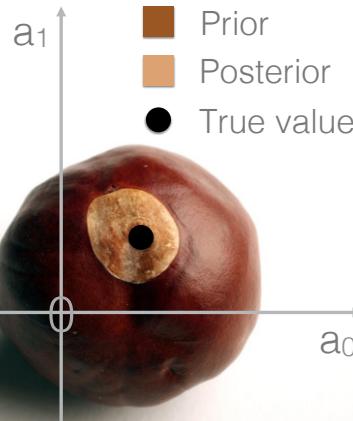
# Bayesian analysis of EFTs with Jupyter notebooks

Dick Furnstahl

Virtual DNP Meeting, October 2020



THE OHIO STATE UNIVERSITY



BUQEYE Collaboration

Special thanks  
to my BUQEYE  
colleagues!

<https://buqeye.github.io>

This talk: [\[pptx\]](#)[\[pdf\]](#)



# What is a Jupyter notebook? (cf. Mathematica notebook)

- Executes Python ([Anaconda](#) recommended; all free!) but also other languages
- Displays in your browser (can run in the cloud through Binder! [Example below](#))
- Tutorials available from [datacamp](#), [dataquest](#), [YouTube](#), ...

## Sampling of 1d pdfs in Python

Here we show how histogrammed samples become closer to the continuous pdf as the sample size increases.

We use a 1d Gaussian distribution:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

In [1]:

```
1 import scipy.stats as stats      # We'll use stats as our source of pdfs
2 import numpy as np              # numpy for arrays
3 import matplotlib.pyplot as plt # matplotlib for plotting
```

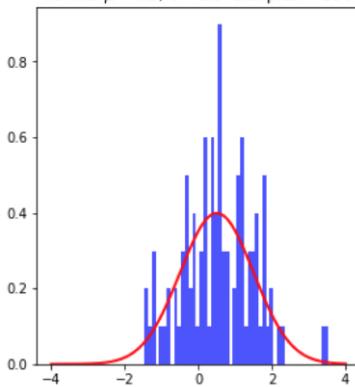
In [2]:

```
1 def plot_hist(ax, name, x_dist, dist, num_samples, num_bins):
2     """Plot a pdf and a histogram of samples"""
3     samples = dist.rvs(size=num_samples) # draw random samples from the pdf
4     # Make a histogram of the random samples
5     count, bins, ignored = ax.hist(samples, num_bins, density=True,
6                                     color='blue', alpha=0.7)
7     ax.plot(x_dist, dist.pdf(x_dist), linewidth=2, color='r') # true pdf
8     title_string = name + f' samples = {num_samples:d}'
9     ax.set_title(title_string)
10
11
12 mu, sigma = 0.5, 1 # mean and standard deviation
13 x_dist = np.linspace(-4, 4, 500) # range of x points
14 name = rf'normal $\mu={mu:1.1f}, $\sigma={sigma:1.1f}' # title for plots
15 num_bins = 50 # histogram bins
```

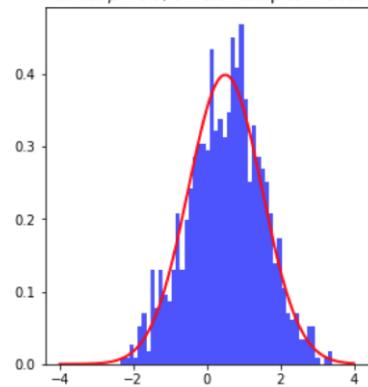
In [3]:

```
1 fig = plt.figure(figsize=(15,5))
2 num_samples = 100
3 norm_dist = stats.norm(mu, sigma)
4 ax_1 = fig.add_subplot(1, 3, 1)
5 plot_hist(ax_1, name, x_dist, norm_dist, num_samples, num_bins)
6
7 num_samples = 1000
8 norm_dist = stats.norm(mu, sigma)
9 ax_2 = fig.add_subplot(1, 3, 2)
10 plot_hist(ax_2, name, x_dist, norm_dist, num_samples, num_bins)
11
```

normal  $\mu=0.5, \sigma=1.0$  samples = 100

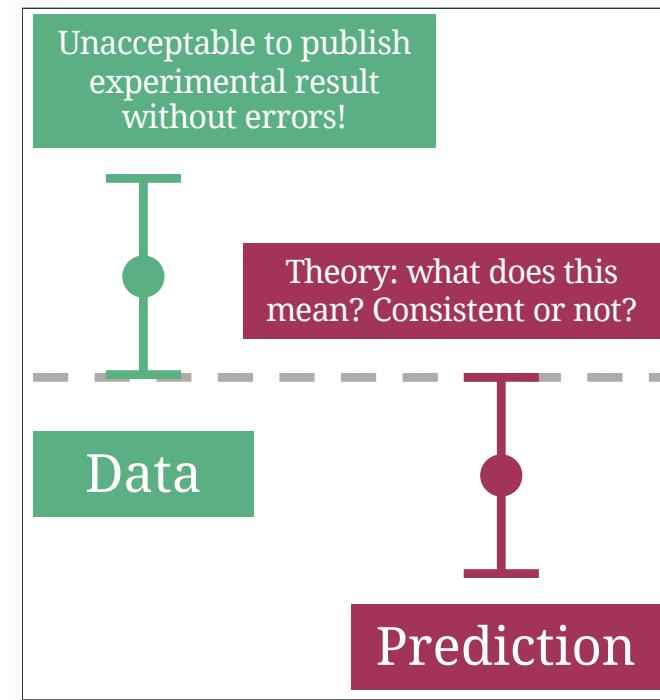
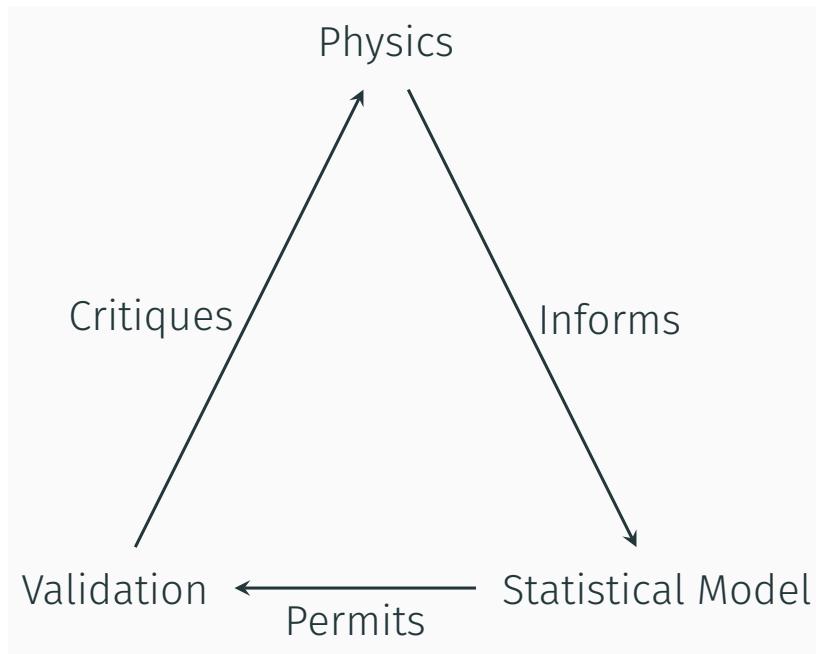


normal  $\mu=0.5, \sigma=1.0$  samples = 1000



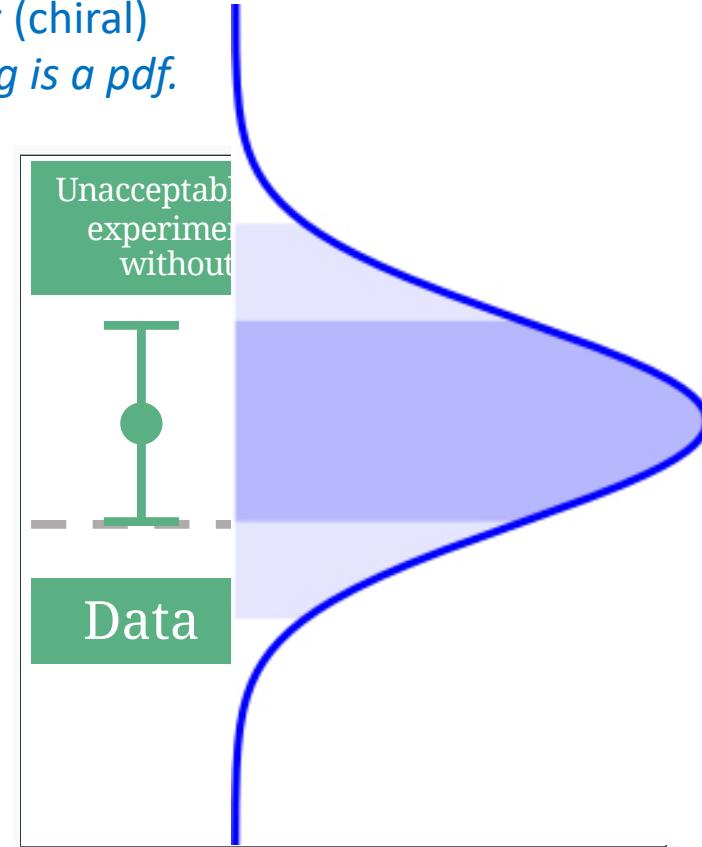
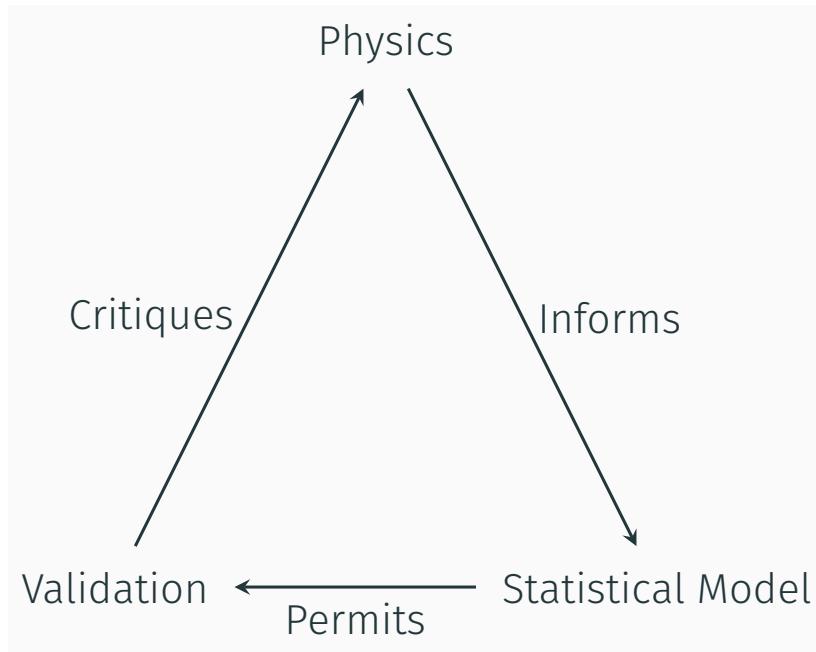
# Overview of Bayesian methods

- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*



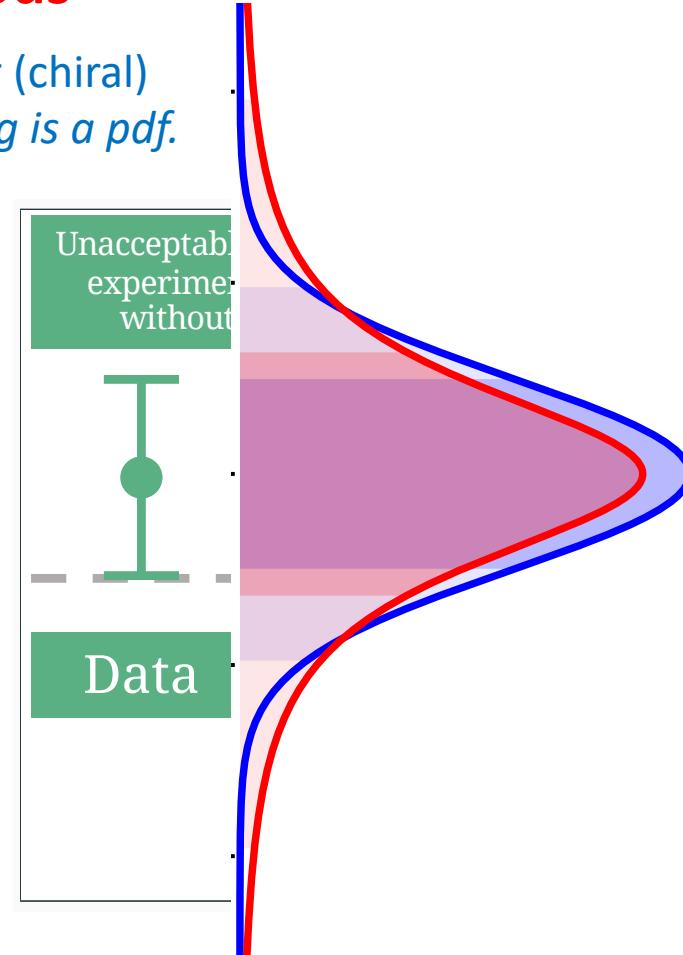
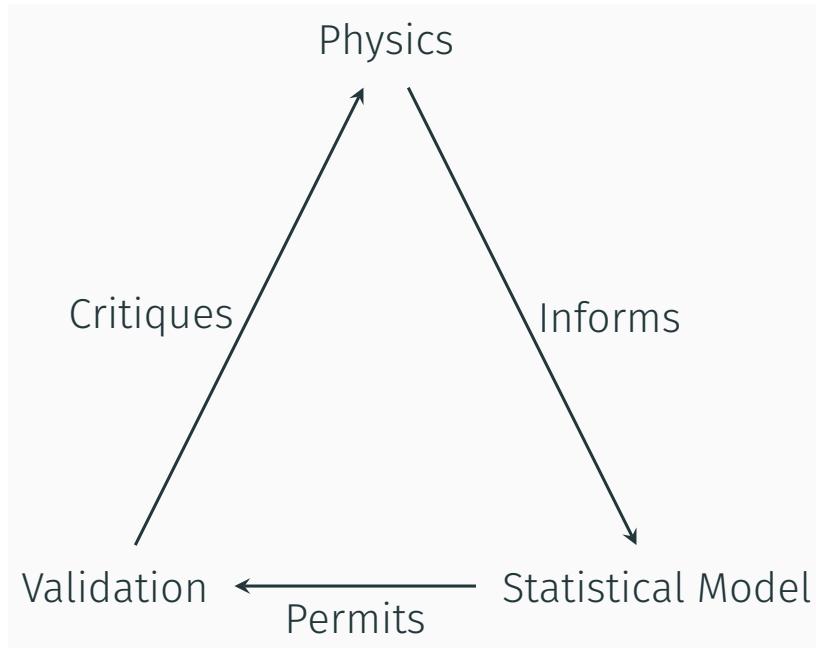
# Overview of Bayesian methods

- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*



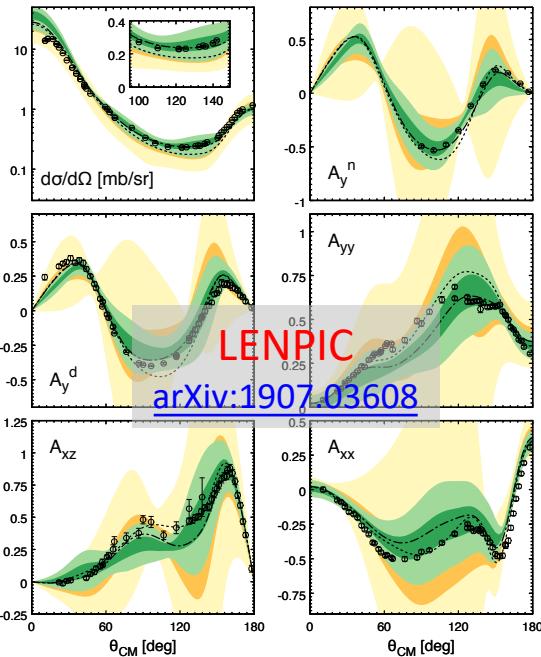
# Overview of Bayesian methods

- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*



# Overview of Bayesian methods

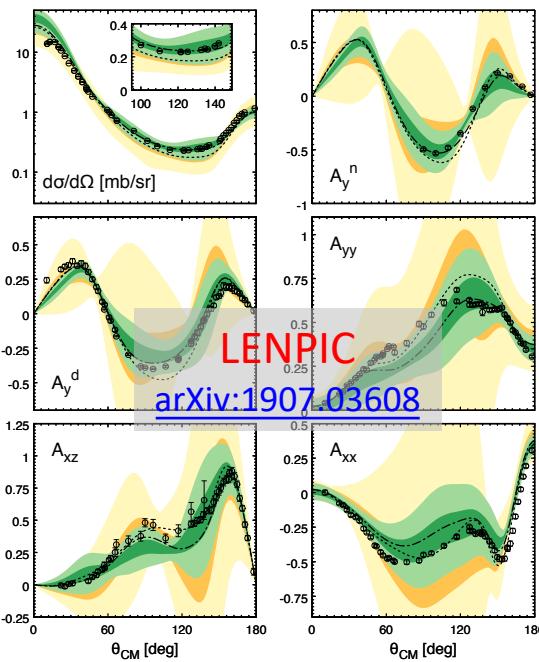
- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*



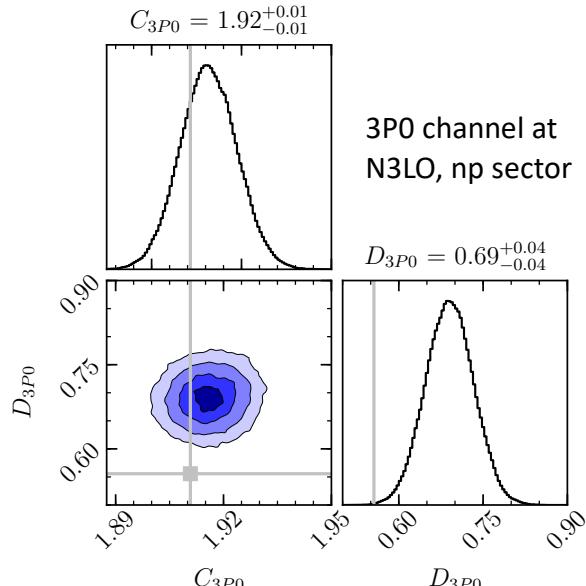
$\text{pr}(\delta \mathbf{y}_{\text{th}} | \mathbf{y}_{\text{th}}, \mathcal{I}) \Rightarrow \text{pdf of theory error}$   
 $\delta \mathbf{y}_{\text{th}}$  given theory calculations  $\mathbf{y}_{\text{th}}$

# Overview of Bayesian methods

- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*



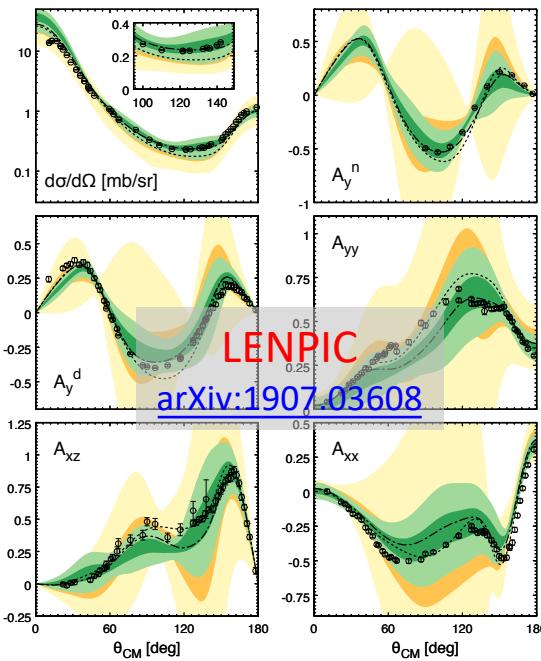
$\text{pr}(\delta \mathbf{y}_{\text{th}} | \mathbf{y}_{\text{th}}, \mathbf{I}) \Rightarrow \text{pdf of theory error}$   
 $\delta \mathbf{y}_{\text{th}}$  given theory calculations  $\mathbf{y}_{\text{th}}$



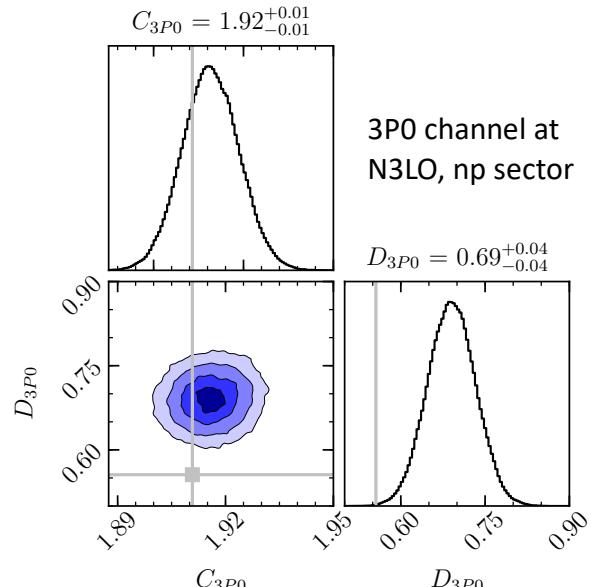
$\text{pr}(\boldsymbol{\theta} | \mathbf{y}_{\text{exp}}, \boldsymbol{\Sigma}_{\text{exp}}, \boldsymbol{\Sigma}_{\text{th}}, \mathbf{I}) \Rightarrow$   
pdf of model parameters  $\boldsymbol{\theta}$  given data  
 $\mathbf{y}_{\text{exp}}$  and experiment/theory errors  $\boldsymbol{\Sigma}$

# Overview of Bayesian methods

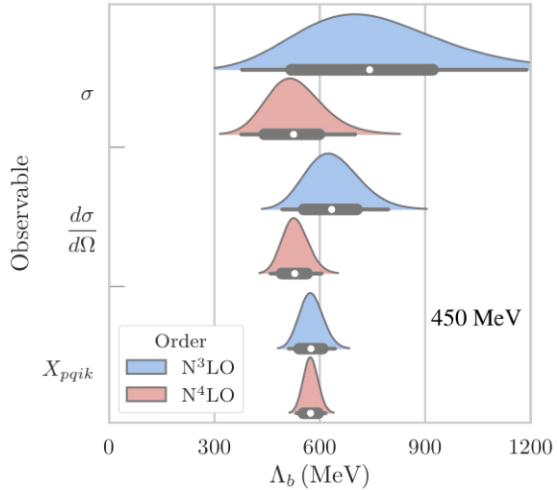
- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*



$pr(\delta y_{th} | y_{th}, I) \Rightarrow$  pdf of theory error  
 $\delta y_{th}$  given theory calculations  $y_{th}$



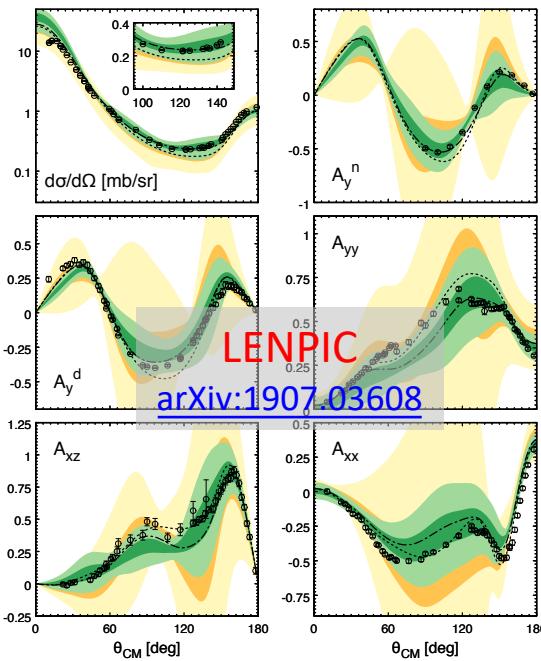
$pr(\theta | y_{exp}, \Sigma_{exp}, \Sigma_{th}, I) \Rightarrow$   
pdf of model parameters  $\theta$  given data  
 $y_{exp}$  and experiment/theory errors  $\Sigma$



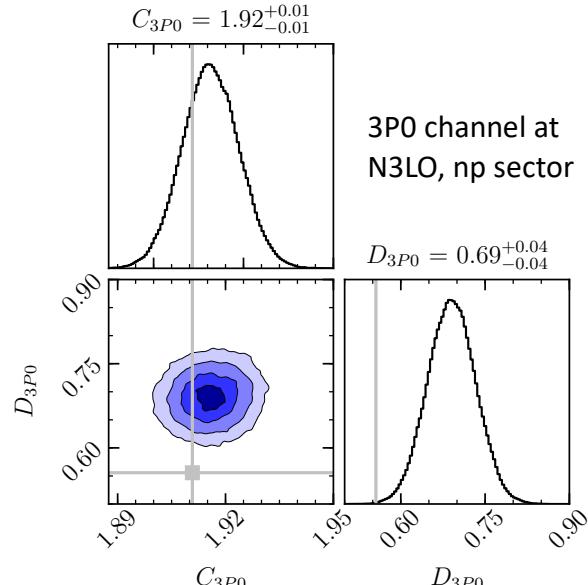
$pr(\Lambda_b | y_{th}, I) \Rightarrow$  pdf of breakdown scale of EFT expansion

# Overview of Bayesian methods

- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*

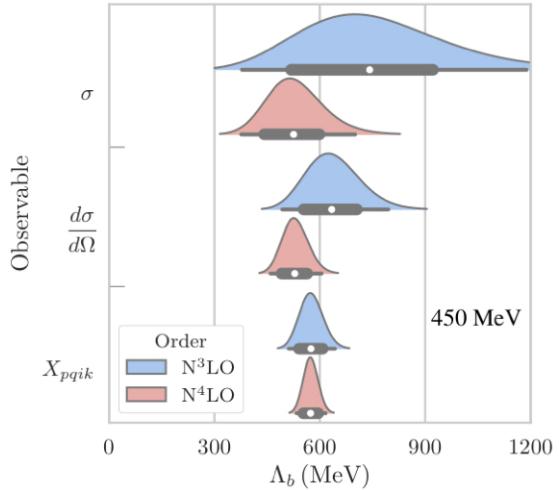


$pr(\delta y_{th} | y_{th}, I) \Rightarrow$  pdf of theory error  
 $\delta y_{th}$  given theory calculations  $y_{th}$

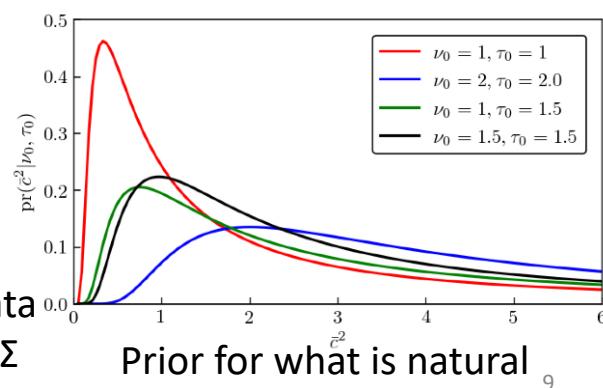


$pr(\theta | y_{exp}, \Sigma_{exp}, \Sigma_{th}, I) \Rightarrow$

pdf of model parameters  $\theta$  given data  
 $y_{exp}$  and experiment/theory errors  $\Sigma$

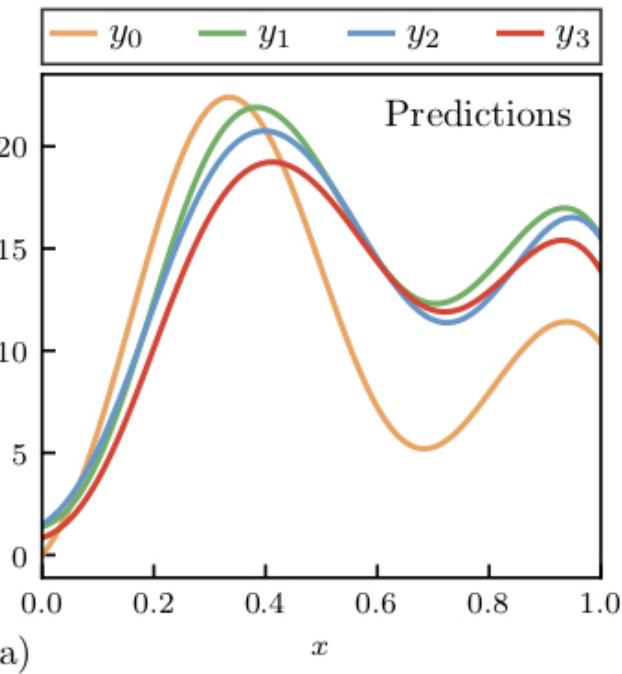
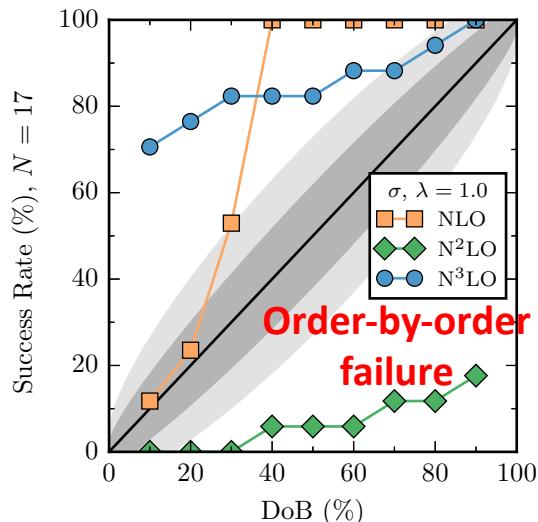
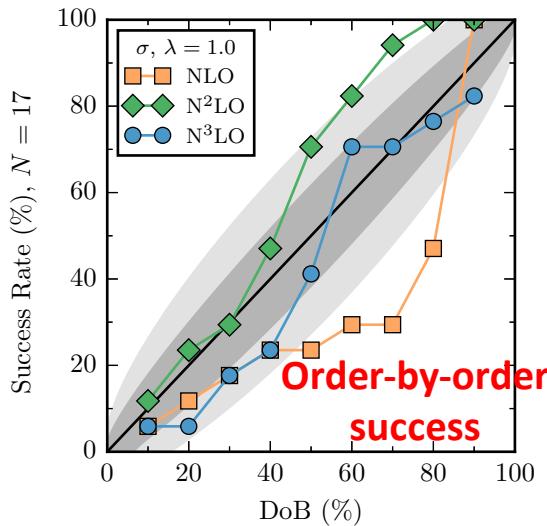


$pr(\Lambda_b | y_{th}, I) \Rightarrow$  pdf of breakdown scale of EFT expansion



# Overview of Bayesian methods

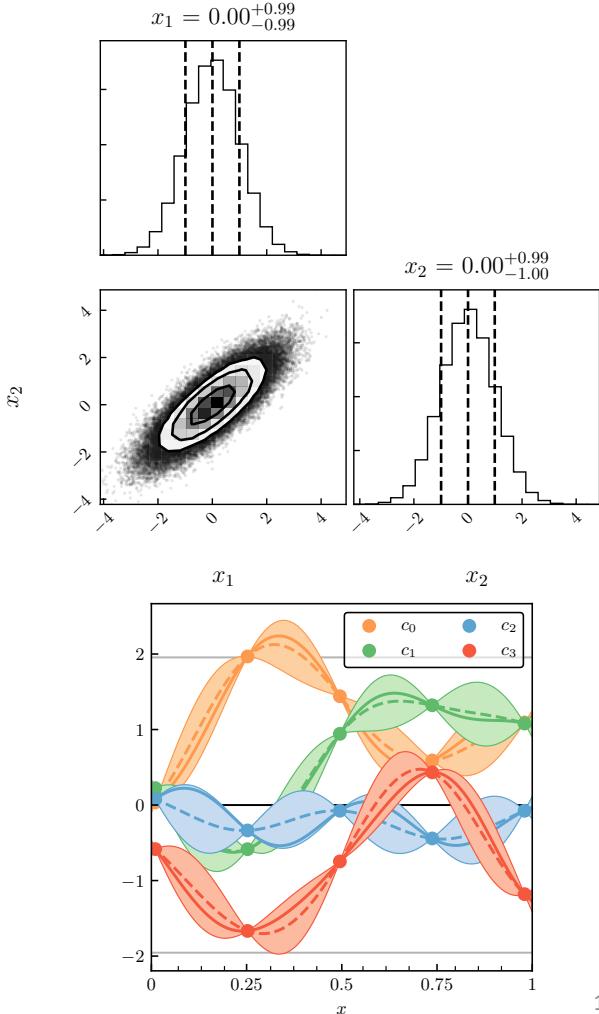
- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*
- EFT theory *discrepancy model* from the convergence pattern and naturalness *priors*; assumptions explicit.
- *Model checking* is an essential part of Bayesian UQ.



Toy model for an observable  $y(x)$ ; e.g., cross section vs. energy, order-by-order in EFT.

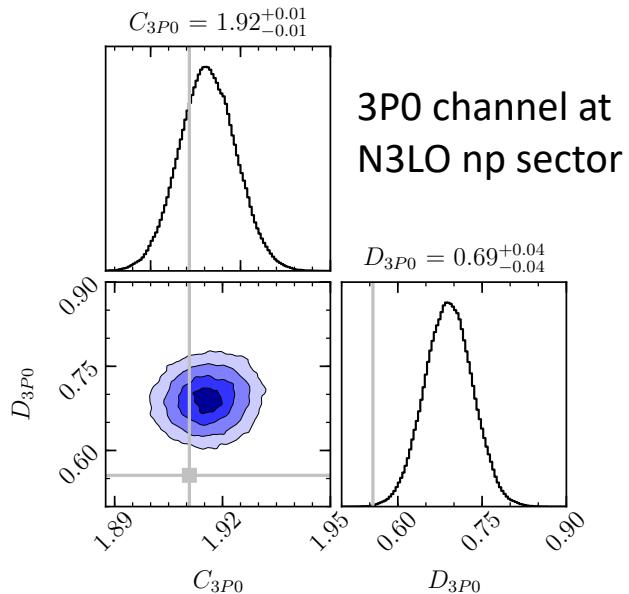
# Overview of Bayesian methods

- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*
- EFT theory *discrepancy model* from the convergence pattern and naturalness *priors*; assumptions explicit.
- *Model checking* is an essential part of Bayesian UQ.
- *Correlations* matter (in many ways). Gaussian process truncation error model for continuous correlations.



# Overview of Bayesian methods

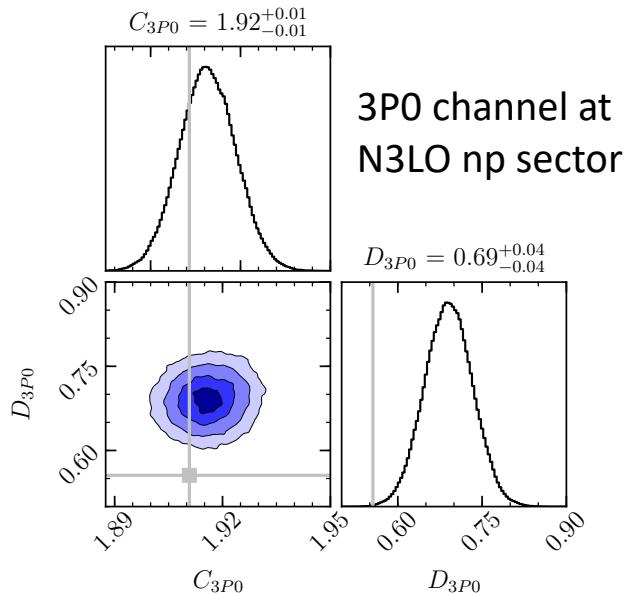
- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*
- EFT theory *discrepancy model* from the convergence pattern and naturalness *priors*; assumptions explicit.
- *Model checking* is an essential part of Bayesian UQ.
- *Correlations* matter (in many ways). Gaussian process truncation error model for continuous correlations.
- Bayesian: *sample* for parameter estimation and the propagation of uncertainties; use *emulators* (like EC)!



$\text{pr}(\boldsymbol{\theta} \mid \mathbf{y}_{\text{exp}}, \Sigma_{\text{exp}}, \Sigma_{\text{th}}, \mathcal{I}) \Rightarrow$   
pdf of model parameters  $\boldsymbol{\theta}$  given data  $\mathbf{y}_{\text{exp}}$  and experiment/theory errors  $\Sigma$

# Overview of Bayesian methods

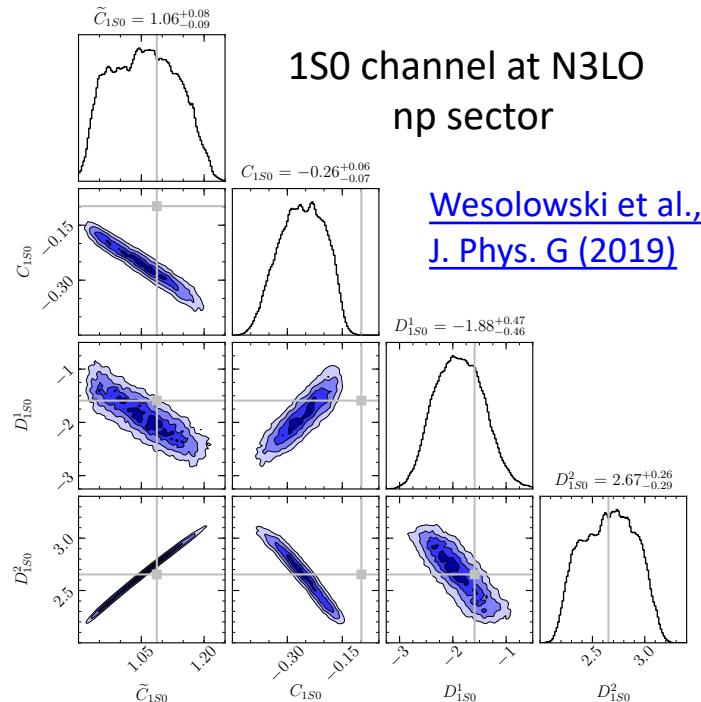
- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*
- EFT theory *discrepancy model* from the convergence pattern and naturalness *priors*; assumptions explicit.
- *Model checking* is an essential part of Bayesian UQ.
- *Correlations* matter (in many ways). Gaussian process truncation error model for continuous correlations.
- **Bayesian: sample** for parameter estimation and the propagation of uncertainties; use *emulators* (like EC)!
- Using priors and truncation errors minimizes overfitting and dependence on how much data is used; posteriors can be used for diagnostics.



$pr(\boldsymbol{\theta} | \mathbf{y}_{\text{exp}}, \Sigma_{\text{exp}}, \Sigma_{\text{th}}, \mathcal{I}) \Rightarrow$   
pdf of model parameters  $\boldsymbol{\theta}$  given data  $\mathbf{y}_{\text{exp}}$  and experiment/theory errors  $\Sigma$

# Overview of Bayesian methods

- Bayesian statistics is a powerful framework for (chiral) EFT uncertainty quantification (UQ). *Everything is a pdf.*
- EFT theory *discrepancy model* from the convergence pattern and naturalness *priors*; assumptions explicit.
- *Model checking* is an essential part of Bayesian UQ.
- *Correlations* matter (in many ways). Gaussian process truncation error model for continuous correlations.
- **Bayesian: sample** for parameter estimation and the propagation of uncertainties; use *emulators* (like EC)!
- Using priors and truncation errors minimizes overfitting and dependence on how much data is used; posteriors can be used for diagnostics.

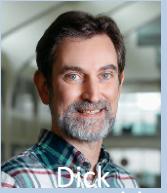


$\text{pr}(\boldsymbol{\theta} \mid \mathbf{y}_{\text{exp}}, \Sigma_{\text{exp}}, \Sigma_{\text{th}}, \mathcal{I}) \Rightarrow$   
pdf of model parameters  $\boldsymbol{\theta}$  given data  $\mathbf{y}_{\text{exp}}$  and experiment/theory errors  $\Sigma$

# BUQEYE Collaboration (“Bayesian Uncertainty Quantification: Errors for Your EFT”)



Christian  
Drischler



Dick  
Furnstahl



Harald  
Grießhammer



Natalie  
Klco



Jordan  
Melendez



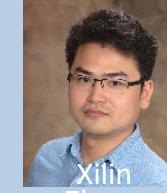
Daniel  
Phillips



Matt  
Pratola

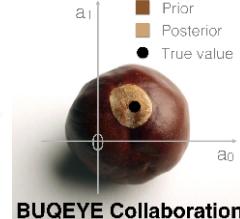


Sarah  
Wesolowski



Xilin  
Zhang

The buckeye is  
the state tree  
of Ohio! 



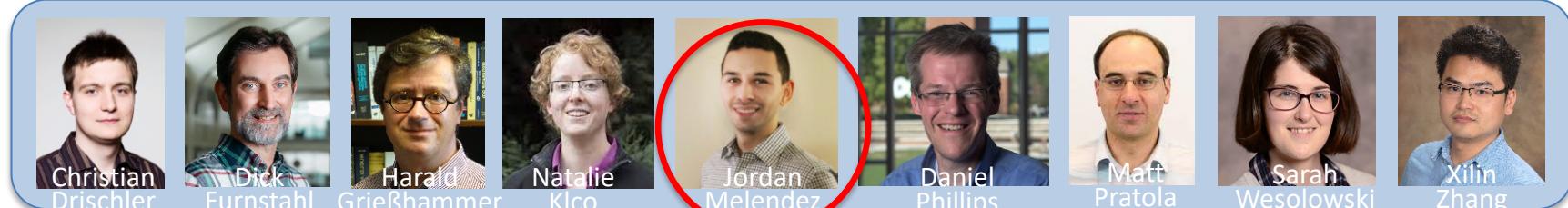
<https://buqeye.github.io>

Papers and software (including Jupyter notebooks for figures)

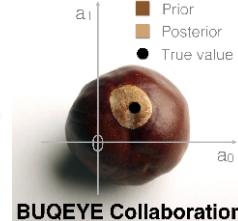
## BUQEYE philosophy for Bayesian analysis via notebooks:

- All assumptions explicit, with model checking and able to be checked externally
- Jupyter notebook(s) that reproduce all figures in BUQEYE papers (data provided)
- Open source, available on Github (or equivalent). Forks are encouraged!

# BUQEYE Collaboration (“Bayesian Uncertainty Quantification: Errors for Your EFT”)



The buckeye is  
the state tree  
of Ohio! →



<https://buqeye.github.io>

Papers and software (including Jupyter notebooks for figures)

## BUQEYE philosophy for Bayesian analysis via notebooks:

- All assumptions explicit, with model checking and able to be checked externally
- Jupyter notebook(s) that reproduce all figures in BUQEYE papers (data provided)
- Open source, available on Github (or equivalent). Forks are encouraged!

Recent highlight: “[Effective Field Theory Truncation Errors and Why They Matter](#),”  
Jordan Melendez PhD thesis [2021 Dissertation Award in Nuclear Physics \(SA.00003 11:42am\)](#)

# The BUQEYE Cheatsheet for Pointwise Truncation Errors (arXiv:1904.10581)

From observable  $y$ , extract coefficients

$$\begin{aligned}\vec{y}_k &\equiv \{y_0, y_1, \dots, y_k\} \\ \Rightarrow \vec{c}_k &\equiv \{c_0, c_1, \dots, c_k\}\end{aligned}\tag{A1}$$

Choose  $\nu_0$  and  $\tau_0$ . Update hyperparameters

$$\nu = \nu_0 + n_c\tag{A7}$$

$$\nu\tau^2 = \nu_0\tau_0^2 + \vec{c}_k^2\tag{A8}$$

Compute posterior

$$\text{pr}(y | \vec{y}_k, Q) \sim t_{\nu} \left[ y_k, y_{\text{ref}}^2 \frac{Q^{2(k+1)}}{1 - Q^2} \tau^2 \right] \tag{A13}$$

```
import numpy as np
y_ref = 20.0; Q = 0.3; k = 3
y_k = [21.7, 27.3, 25.4, 26.2]
c_k = np.array([y_k[0] / y_ref] + [
    (y_k[n] - y_k[n-1]) / (y_ref * Q**n)
    for n in range(1, k+1)])
nu_0 = 1; tau_0 = 1 # ~Uninformative
nu = nu_0 + len(c_k)
tau_sq = \
    (nu_0 * tau_0**2 + c_k @ c_k) / nu

from scipy.stats import t
scale = y_ref * Q**(k+1) * \
    (tau_sq / (1 - Q**2))**0.5
y = t(nu, y_k[-1], scale)
dob = y.interval(0.95) # (25.7, 26.7)
```

Note: If  $n_c \gg 1$ , the posterior for  $y$  becomes a normal distribution.

From <https://buqeye.github.io/>

From observable  $y$ , extract coefficients

$$\begin{aligned}\vec{y}_k &\equiv \{y_0, y_1, \dots, y_k\} \\ \Rightarrow \vec{c}_k &\equiv \{c_0, c_1, \dots, c_k\}\end{aligned}\quad (\text{A1})$$

$$y_k = y_{\text{ref}} \sum_{n=0}^k c_n Q^n \quad \rightarrow \quad \delta y_{\text{th}} = y_{\text{ref}} \sum_{n=k+1}^{\infty} c_n Q^n$$

$$\chi\text{EFT} \Rightarrow Q = \frac{\{p, m_\pi\}}{\Lambda_b}, \quad \Lambda_b \approx 600 \text{ MeV}$$

Compute posterior

$$\text{pr}(y | \vec{y}_k, Q) \sim t_\nu \left[ y_k, y_{\text{ref}}^2 \frac{Q^{2(k+1)}}{1 - Q^2} \tau^2 \right] \quad (\text{A13})$$

```
import numpy as np
y_ref = 20.0; Q = 0.3; k = 3
y_k = [21.7, 27.3, 25.4, 26.2]
c_k = np.array([y_k[0] / y_ref] + [
    (y_k[n] - y_k[n-1]) / (y_ref * Q**n)
    for n in range(1, k+1)])
nu_0 = 1; tau_0 = 1 # ~Uninformative
nu = nu_0 + len(c_k)
tau_sq = \
    (nu_0 * tau_0**2 + c_k @ c_k) / nu

from scipy.stats import t
scale = y_ref * Q**(k+1) * \
    (tau_sq / (1 - Q**2))**0.5
y = t(nu, y_k[-1], scale)
dob = y.interval(0.95) # (25.7, 26.7)
```

Note: If  $n_c \gg 1$ , the posterior for  $y$  becomes a normal distribution.

From <https://buqeye.github.io/>

From observable  $y$ , extract coefficients

$$\begin{aligned}\vec{y}_k &\equiv \{y_0, y_1, \dots, y_k\} \\ \Rightarrow \vec{c}_k &\equiv \{c_0, c_1, \dots, c_k\}\end{aligned}\quad (\text{A1})$$

Choose  $\nu_0$  and  $\tau_0$ . Update hyperparameters

$$\nu = \nu_0 + n_c \quad (\text{A7})$$

$$\nu\tau^2 = \nu_0\tau_0^2 + \vec{c}_k^2 \quad (\text{A8})$$

Compute posterior

$$\text{pr}(y | \vec{y}_k, Q) \sim t_{\nu} \left[ y_k, y_{\text{ref}}^2 \frac{Q^{2(k+1)}}{1 - Q^2} \tau^2 \right] \quad (\text{A13})$$

```
import numpy as np
y_ref = 20.0; Q = 0.3; k = 3
y_k = [21.7, 27.3, 25.4, 26.2]
c_k = np.array([y_k[0] / y_ref] + [
    (y_k[n] - y_k[n-1]) / (y_ref * Q**n)
    for n in range(1, k+1)])
nu_0 = 1; tau_0 = 1 # ~Uninformative
nu = nu_0 + len(c_k)
tau_sq = \
    (nu_0 * tau_0**2 + c_k @ c_k) / nu
from scipy.stats import t
scale = y_ref * Q**(k+1) * \
    (tau_sq / (1 - Q**2))**0.5
y = t(nu, y_k[-1], scale)
dob = y.interval(0.95) # (25.7, 26.7)
```

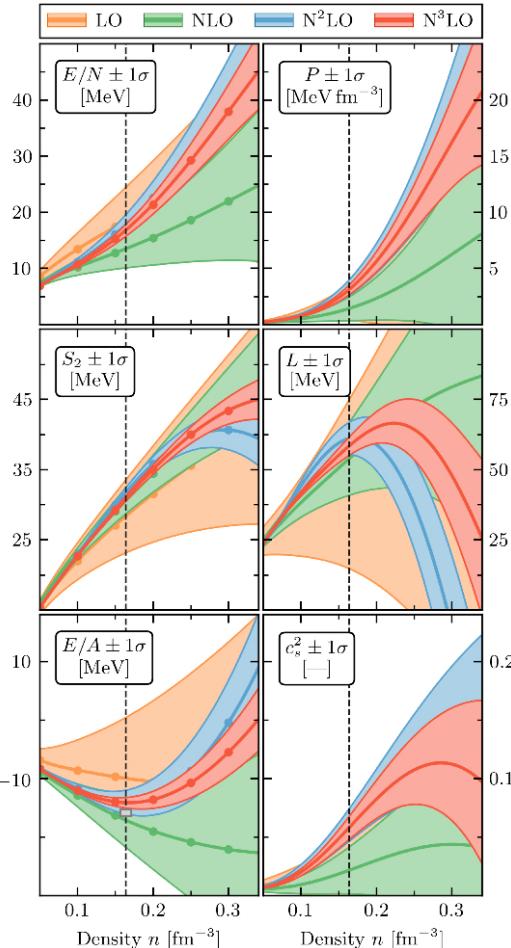
Note: If  $n_c \gg 1$ , the posterior for  $y$  becomes a normal distribution.

From <https://buqeye.github.io/>

# Recent BUQEYE notebooks for EFT UQ papers

- **Quantifying correlated truncation errors in effective field theory**, J. Melendez, rjf, D. Phillips, M. Pratola, and S. Wesolowski, [Phys. Rev. C 100, 044001 \(2019\)](#) [\[notebooks\]](#)
- **How well do we know the neutron-matter equation of state at the densities inside neutron stars? A Bayesian approach with correlated uncertainties**, C. Drischler, rjf, J. Melendez, and D. Phillips, [Phys. Rev. Lett. \(in press\)](#). [\[notebooks\]](#)
- **Quantifying uncertainties and correlations in the nuclear-matter equation of state**, C. Drischler, J. Melendez, rjf, and D Phillips, [Phys. Rev. C \(in press\)](#). [\[notebooks\]](#)
- **Efficient emulators for scattering using eigenvector continuation**, rjf, A. Garcia, P. Millican, and X. Zhang, [Phys. Lett. B 809, 135719 \(2020\)](#). [\[notebooks\]](#)
- **Designing Optimal Experiments: An Application to Proton Compton Scattering**, J. Melendez, rjf, H. Grießhammer, J. McGovern, D. Phillips, and M.T. Pratola, [arXiv:2004.11307](#). [\[notebooks\]](#)  
See Harold Grießhammer talk following this one.

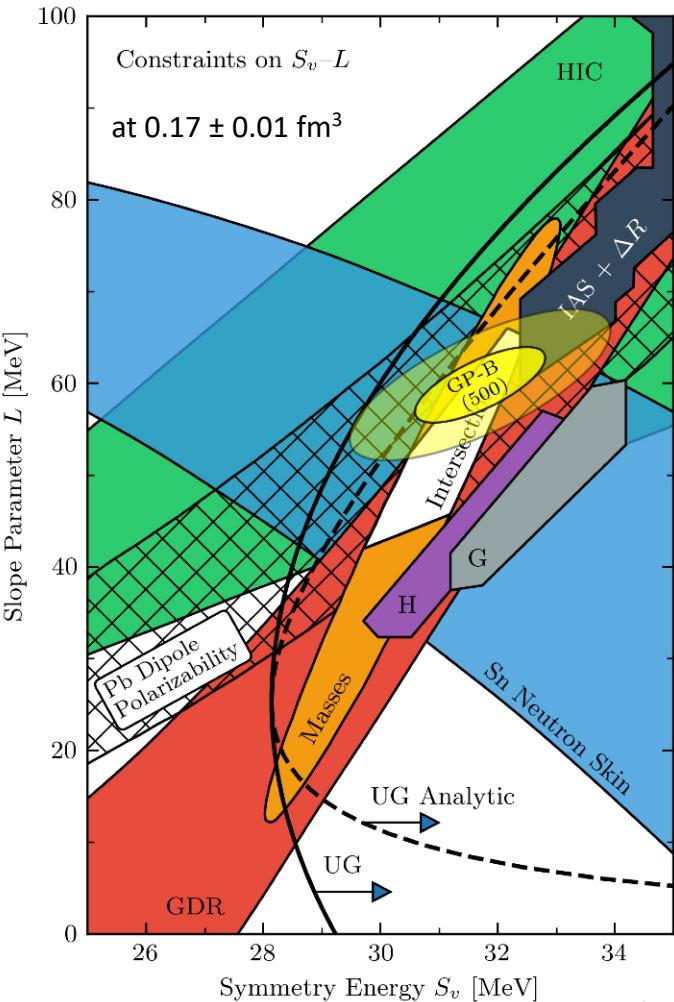
# Correlated theory errors for EOS properties



C. Drischler et al.  
(arXiv:2004.07232  
PRL, in press)  
(SD.00006 11:30am)

Correlated GP treatment  
gives better estimates  
for truncation errors  
and clean propagation  
of uncertainties to  
derived quantities.

See also comparisons to  
GW and NICER posteriors!



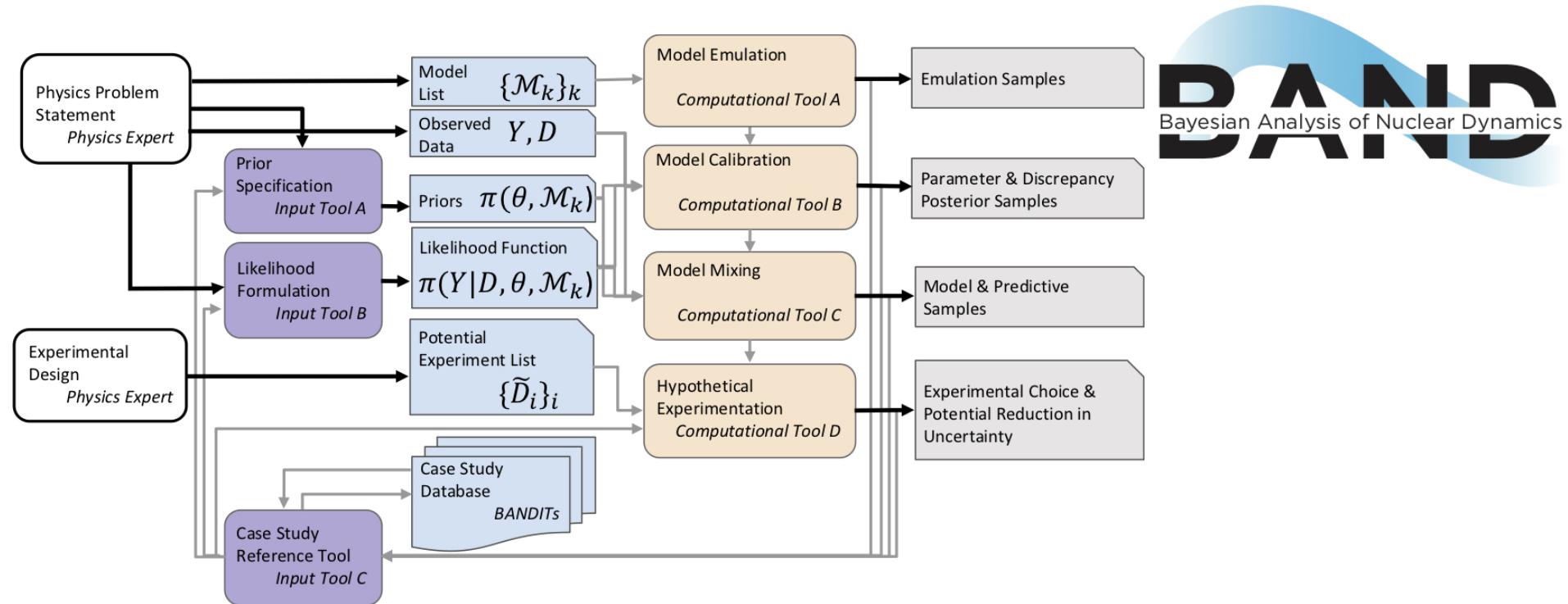
# Summary points and more information

- Jupiter notebooks and Python are great tools for nuclear physics UQ
- Bayesian methods for EFT and other theory errors (combined with experiment)
  - Many examples from the BUQEYE collaboration
- We strongly advocate that *every* paper have a notebook for reproducing figures
- Github repositories with notebooks for learning Bayesian statistics for physics
  - BAYES 2019 (TALENT course): <https://nucleartalent.github.io/Bayes2019/>
  - Furnstahl course at Ohio State: <https://furnstahl.github.io/Physics-8805/>

# BAND (Bayesian Analysis of Nuclear Dynamics)

An NSF Cyberinfrastructure for Sustained Scientific Innovation (CSSI) Framework (from 7/2020)

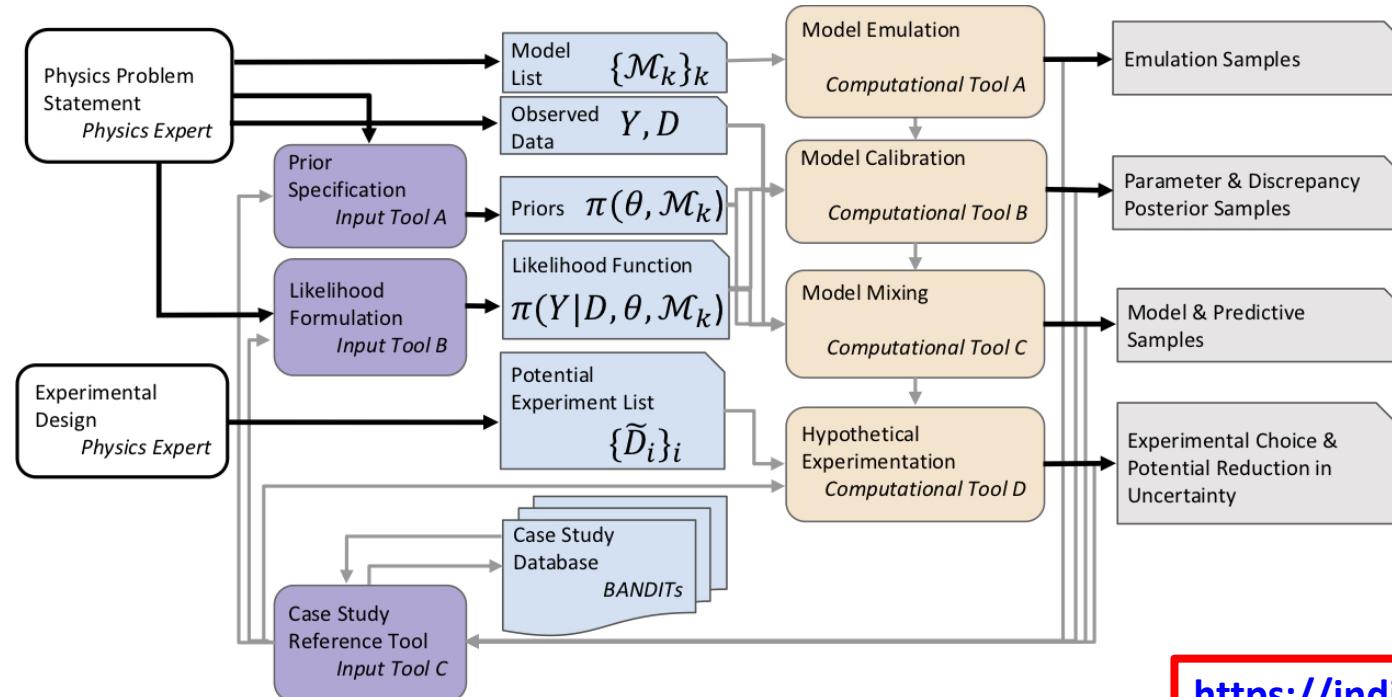
Look to <https://bandframework.github.io/> over the next five years!



# BAND (Bayesian Analysis of Nuclear Dynamics)

An NSF Cyberinfrastructure for Sustained Scientific Innovation (CSSI) Framework (from 7/2020)

Look to <https://bandframework.github.io/> over the next five years!



# BAND

Bayesian Analysis of Nuclear Dynamics

Registration for  
Virtual ISNET 8 and  
for the First Annual  
BAND Camp is open.  
The website lists  
confirmed speakers  
for both events.

<https://indico.frib.msu.edu/event/21/>

Thank you!