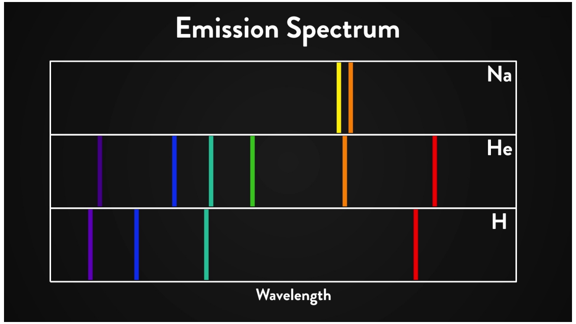
Model Building with Non-Parametric and Parametric Components for Partial Wave Analysis

Jefferson Lab

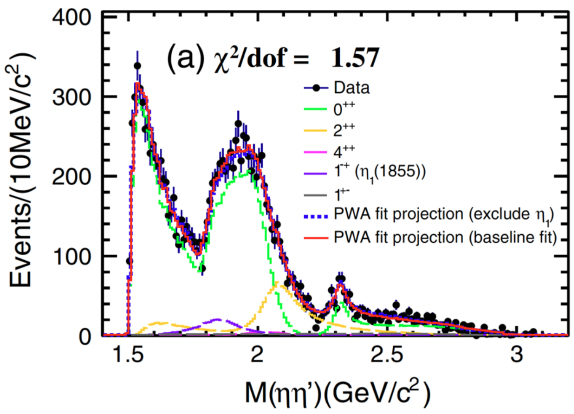
Lawrence Ng
CHEP 2024

GlueX

Hadron Spectroscopy + Partial Wave Analysis



Quantum Mechanics
QED



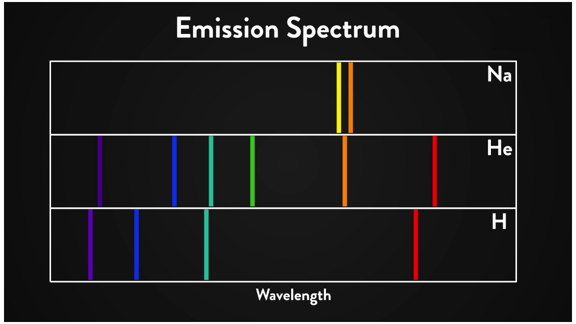
Phys. Rev. Lett. **129**, 192002

Low Energy QCD

Degrees of freedom

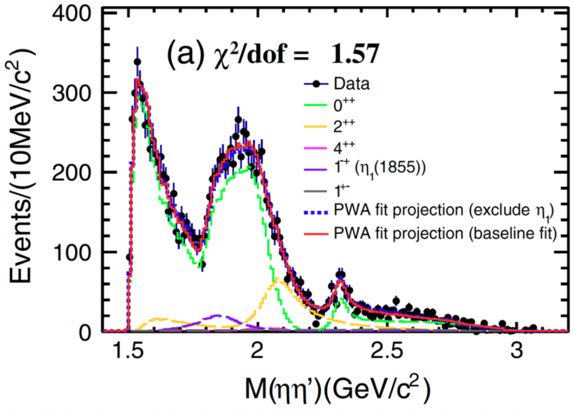
- Baryons / Mesons
- "Exotic" configurations
  - Tetraquarks
  - Pentaquarks
  - Glueballs
  - Hybrid mesons

Hadron Spectroscopy + Partial Wave Analysis



QED

Degrees of Freedom:
Atoms / Molecules



Phys. Rev. Lett. **129**, 192002

QCD

Broad overlapping resonances

- Partial wave analysis (generalized Fourier analysis) to separate **interfering complex-valued** contributions
- **Complicated dynamics that can be hard to model**

Modeling the Complicated Dynamics

Mass Independent Fits

Pros:

- Minimize model dependence

Cons:

- Prone to instabilities from:
  - Ambiguities
  - Numerical (lower stats)

Largely unexplored

Mass Dependent Fits

Pros:

- Smooth results by construction
- Assume some physics (i.e. extract resonance parameters)

Cons:

- Biased results / heuristics

Can we (prior)itize smooth dynamics without specifying functional forms?

Yes, but first we need some core concepts

Base Knowledge 1/2: **Gaussian Processes**

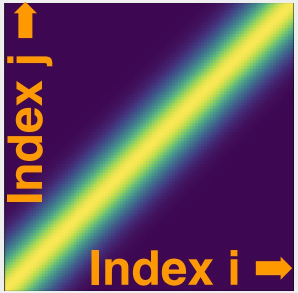
- Generalization of Multivariate Gaussian to infinite dimensions
- At the core: **Kernel Function**

$$\kappa(x_i, x_j) = Cov(X, X') = \Sigma$$

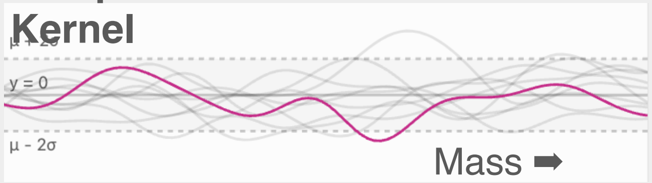
  - Similarity measure / covariance between two points

Radial Basis Function

$$\sigma^2 \exp\left(-\frac{||t-t'||^2}{2l^2}\right)$$

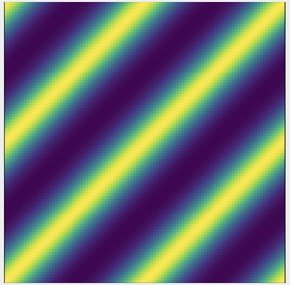


Samples drawn from Kernel

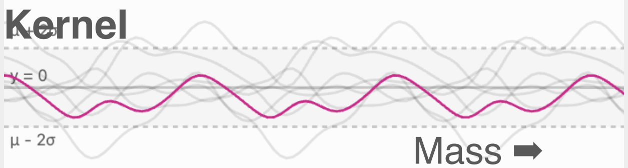


Periodic Kernel

$$\sigma^2 \exp\left(-\frac{2\sin^2(\pi|t-t'|/p)}{l^2}\right)$$



Samples drawn from Kernel



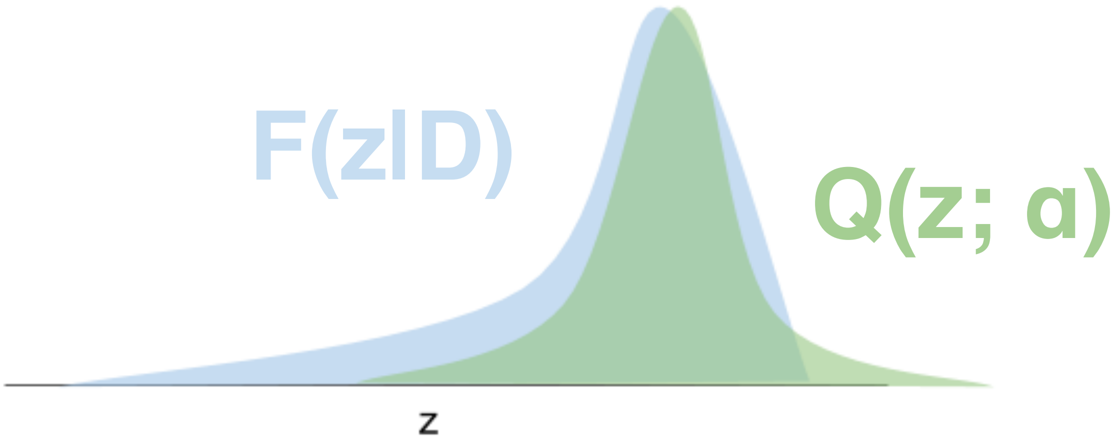
Specific Kernels are chosen based on domain knowledge

*But! We **can** also learn the kernel from data!*

[Distill.pub: A Visual Exploration of Gaussian Processes](#)

Base Knowledge 2/2: **Variational Inference**

F(z|D) = Complicated Posterior Function
Q(z; α) = Simple function
Vary α such that Q(z; α) ≈ F around some point



Numerical Information Field Theory

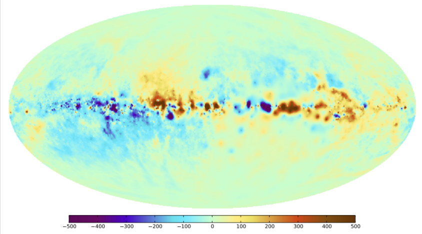
Inference Framework developed for astrophysics at Max Planck Institute for Astrophysics

G. Edenhofer, P. Frank, J. Roth, R. H. Leike, M. Guerdi, L. I. Scheel-Platz, M. Guardiani, V. Eberle, M. Westerkamp, and T. A. Enßlin. Re-Envisioning Numerical Information Field Theory (NIFTy.re): A Library for Gaussian Processes and Variational Inference, 2024.

Mainly working with:
Philipp Frank, Torsten Enßlin, Jakob Knollmüller

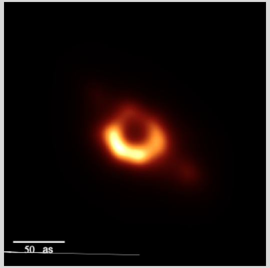
Description

NIFTy, "**N**umerical **I**nformation **F**ield **T**heory", is a Bayesian ~~imaging~~ library.
It is designed to infer the million to billion dimensional posterior distribution ~~in the image space~~ from noisy input data. At the core of NIFTy lies a set of powerful Gaussian Process (GP) models and accurate Variational Inference (VI) algorithms.



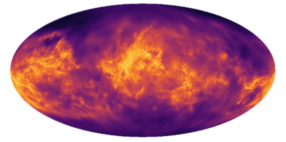
An improved map of the Galactic Faraday sky

N. Oppermann*[1], H. Junklewitz[1], G. Robbers[1], M.R. Bell[1], T.A. Enßlin[1], A. Bonafede[2], R. Braun[3], J.C. Brown[4], T.E. Clarke[5], I.J. Feain[3], B.M. Gaensler[6], A. Hammond[6], L. Harvey-Smith[3], G. Heald[7], M. Johnston-Hollitt[8], U. Klein[9], P.P. Kronberg[10,11], S.A. Mao[3,12], N.M. McClure-Griffiths[3], S.P. O'Sullivan[3], L. Pratley[8], T. Robishaw[13], S. Roy[14], D.H.F.M. Schnitzeler[3,15], C. Sotomayor-Beltran[16], J. Stevens[3], J.M. Stil[4], C. Sunstrum[4], A. Tanna[17], A.R. Taylor[4], and C.L. Van Eck[4]



Variable structures in M87* from space, time and frequency resolved interferometry

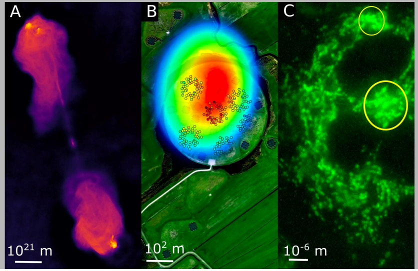
Philipp Arras[1,2], Philipp Frank[1,3], Philipp Haim[1], Jakob Knollmüller[1,2], Reimar Leike[1], Martin Reinecke[1], and Torsten Enßlin[1]



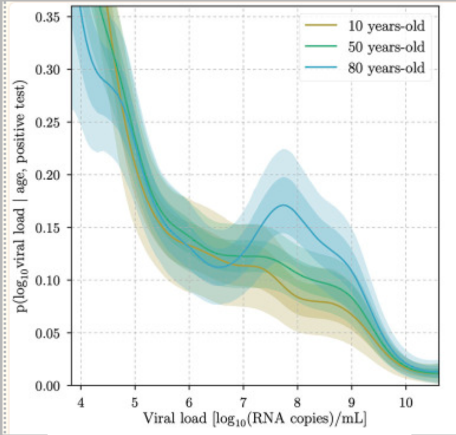
Resolving nearby dust clouds*

R. H. Leike[1,2], M. Glatzle[1,3], and T. A. Enßlin[1,2]

Astrophysics



Radiation biology, radio astronomy and cosmic rays using information field theory

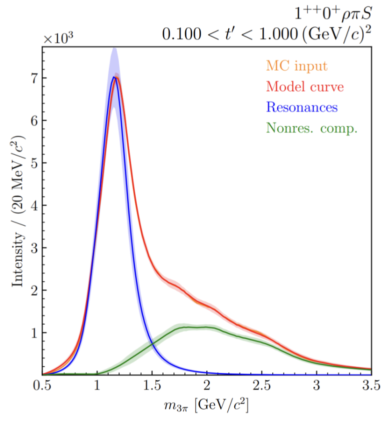


Causal, Bayesian, & non-parametric modeling of the SARS-CoV-2 viral load distribution vs. patient's age

Matteo Guardiani[1,2*], Philipp Frank[1,2], Andrija Kostić[1,2], Gordian Edenhofer[1,2], Jakob Roth[1,2], Berit Uhlmann[4], Torsten Enßlin[1,2,3]

Biology

NIFTy



Hadron Physics?

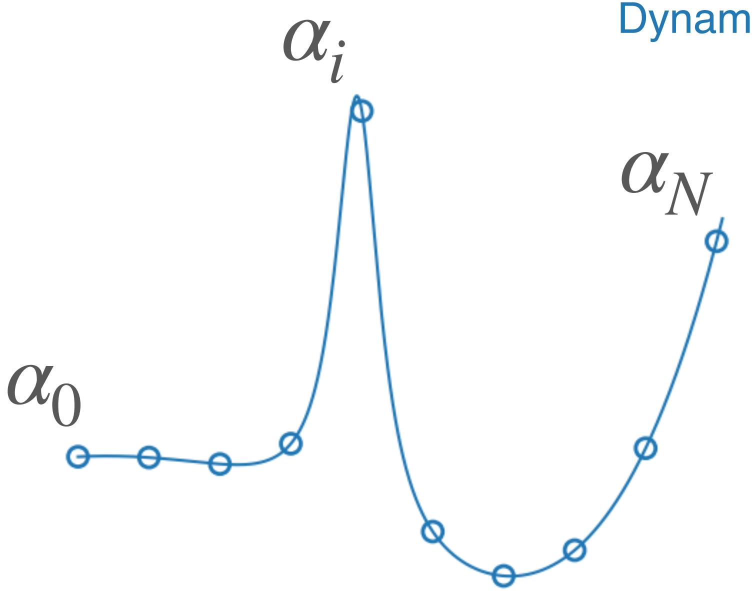
$$\mathcal{I}(\alpha, \tau) \propto |\mathcal{M}(\alpha, \tau)|^2$$

α ~ Production Kinematics
τ ~ Decay Kinematics

$$\mathcal{M}(\alpha, \tau) = \mathcal{C}(\alpha)\mathcal{P}(\alpha)\Psi(\tau)$$

Production **Propagation** **Decay**

Dynamics largely unknown ➡ Bin, assuming constant



Mass-independent (Traditional) Fit

For each kinematic bin:

- Independently maximize likelihood

$$P_i(D_i|\alpha_i)$$

- Can lead to non-smooth results
- Exploit bin-to-bin correlations to obtain smoother results

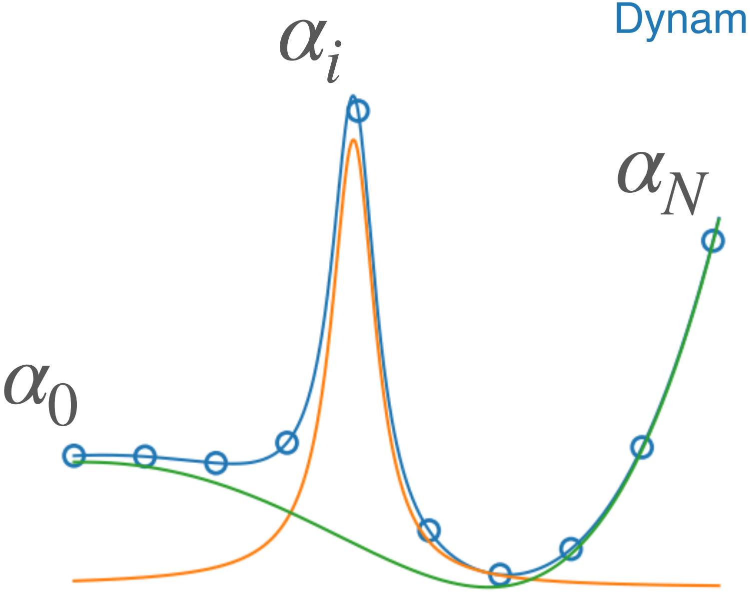
$$\mathcal{I}(\alpha,\tau) \propto |\mathcal{M}(\alpha,\tau)|^2$$

α ~ Production Kinematics
τ ~ Decay Kinematics

$$\mathcal{M}(\alpha,\tau) = \mathcal{C}(\alpha)\mathcal{P}(\alpha)\Psi(\tau)$$

Production **Propagation** **Decay**

Dynamics largely unknown ➡ Bin, assuming constant



Mass-independent (Traditional) Fit

For each kinematic bin:

- Independently maximize likelihood

$$P_i(D_i|\alpha_i)$$

- Can lead to non-smooth results
- Exploit bin-to-bin correlations to obtain smoother results
- Better extract resonance content?

Proposed Method using NIFTy

Sum over kinematically binned likelihoods

$$\sum_i P_i(D_i|\alpha_i)$$

$$P(D|\alpha)$$

Already running PWA fits with O(100k) free parameters

Introduce Prior on α

Gaussian Process Prior to describe smoothness (more later)

+

Priors for Parametric Model (i.e. Breit-Wigner resonance parameters)

$$P(D|\alpha)P(\alpha)$$

$$P(\alpha|D) \propto P(D|\alpha)P(\alpha)$$

Variationally Approximate | Algorithms: MGVI / geoVI

Software: `iftpwa`

Likelihood Manager

Can we develop inter-collaboration software?

python™ OPEN MPI

<u>Autograd</u>
Gradients + Hessians in each bin

<u>Introduce Prior on α</u>

Gaussian Process Prior to describe smoothness

\+

Priors for Parametric Model (i.e. Breit-Wigner resonance parameters)

`iftpwa` is a toolkit of **priors** and **physics functions** allowing creation of **complex-valued** models mixing **parametric** and **non-parametric** components for **variational inference** using `Nifty`

FM Kaspar et al., EPJ Web Conf., 291 (2024) 02014

Gaussian Process Prior

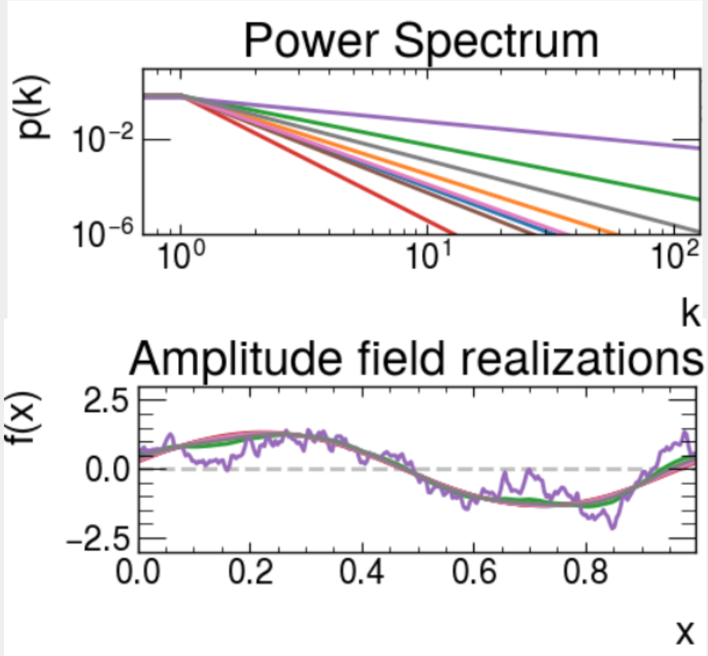
- **Kernels** are defined in **Fourier Space** whose parameters are **Log-Normally Distributed**

Example:

Power spectrum slope parameter ~ LogNormal(-6, 3)

Draw 8 samples

[NIFTy Correlated Field Demo]

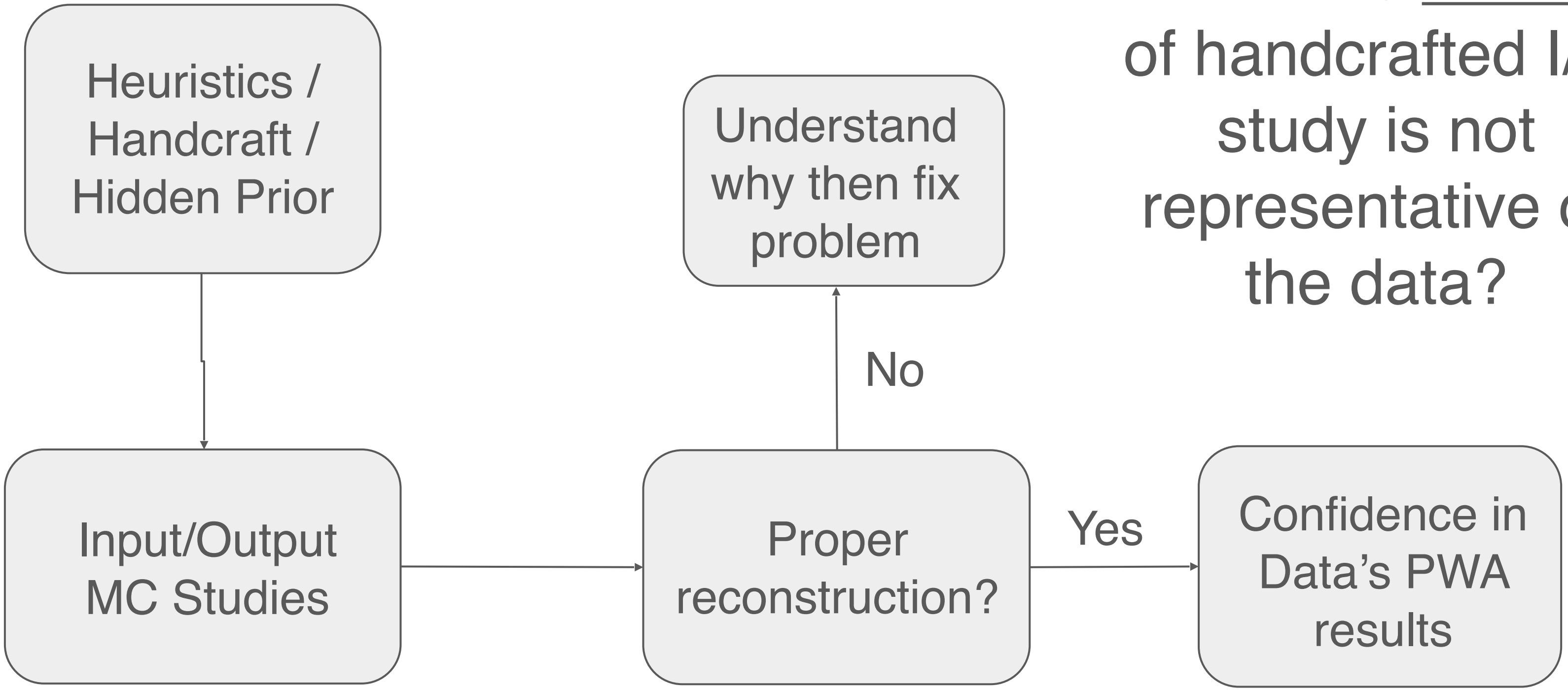


Other knobs:

Constant offsets

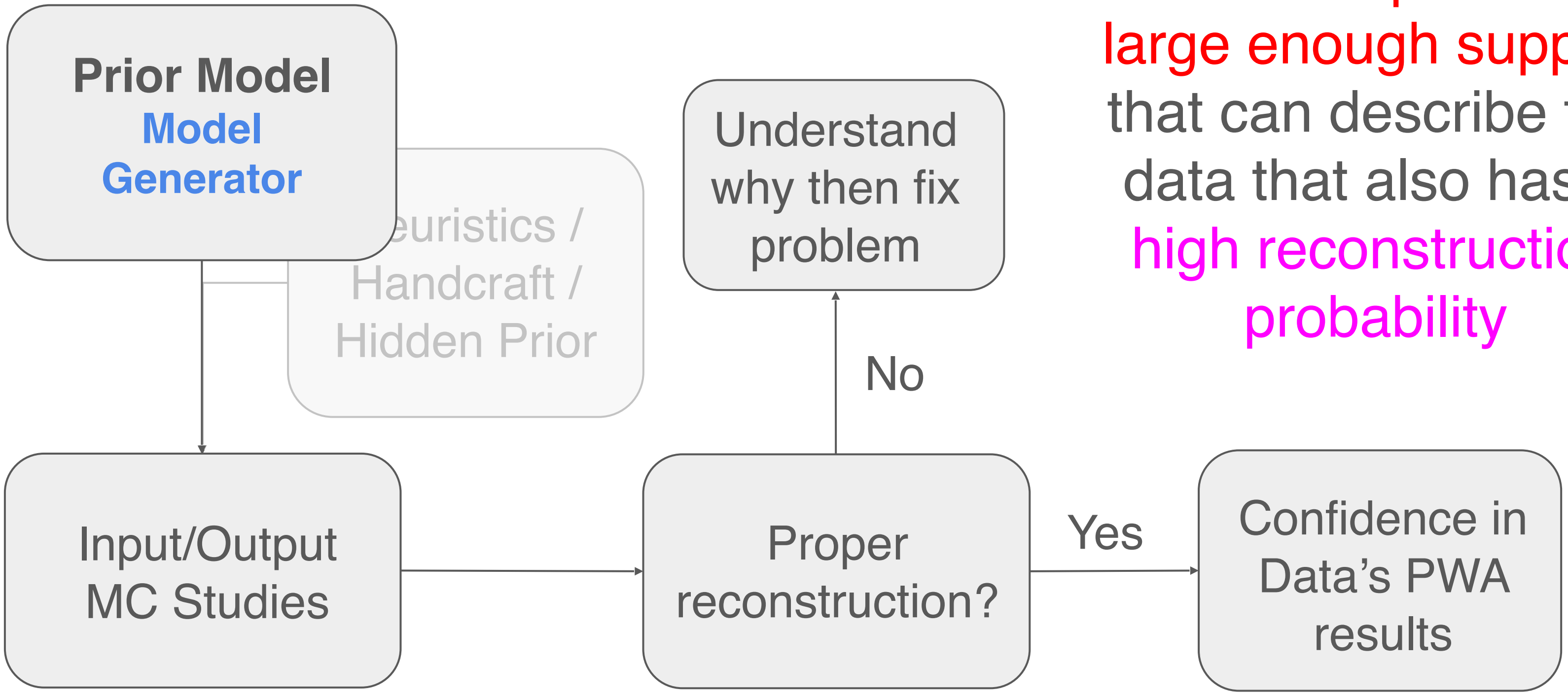
Deform spectrum beyond simple power law

I/O Tests: Typical Procedure



What if the **handful** of handcrafted I/O study is not representative of the data?

I/O Tests: Bayesian Approach

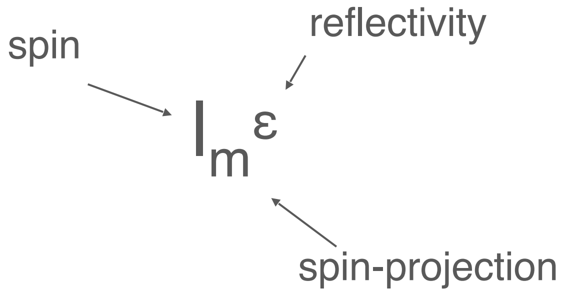


Construct a prior with a large enough support that can describe the data that also has a high reconstruction probability

Input / Output Tests

- Draw a sample from the prior
- Generate events with the sampled functional form of the amplitude
- Fit the events using
  1) Traditional binned maximum likelihood
  2) `iftpwa` framework

- Polarized photoproduction of two pseudoscalar : **$\gamma p \rightarrow \eta\pi^0 p \rightarrow 4\gamma p$**
  - Amplitudes described in: **[V.Mathieu et.al. (JPAC), Phys.Rev.D 100 (2019) 5, 054017]**
  - Form:

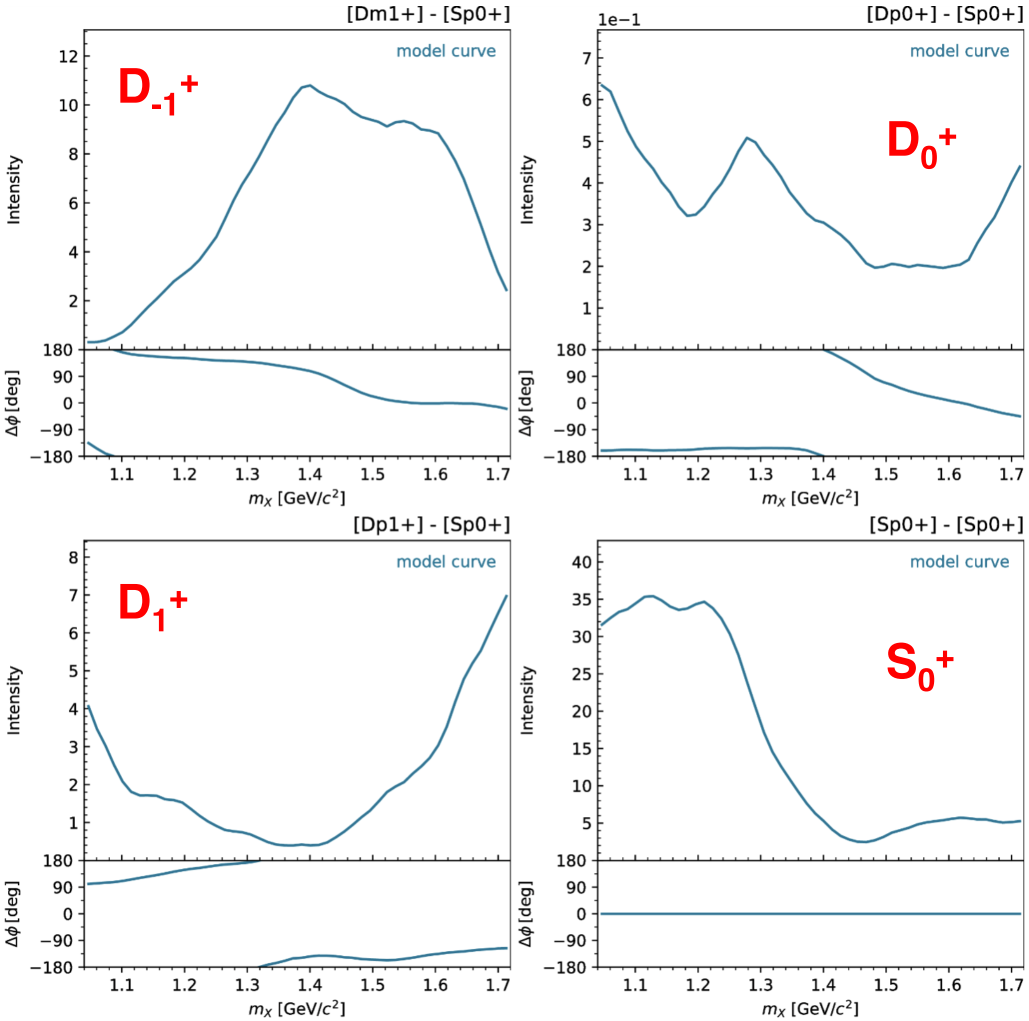


I/O Study 1

**No physics, no resonances,
only arbitrary but smooth amplitudes**

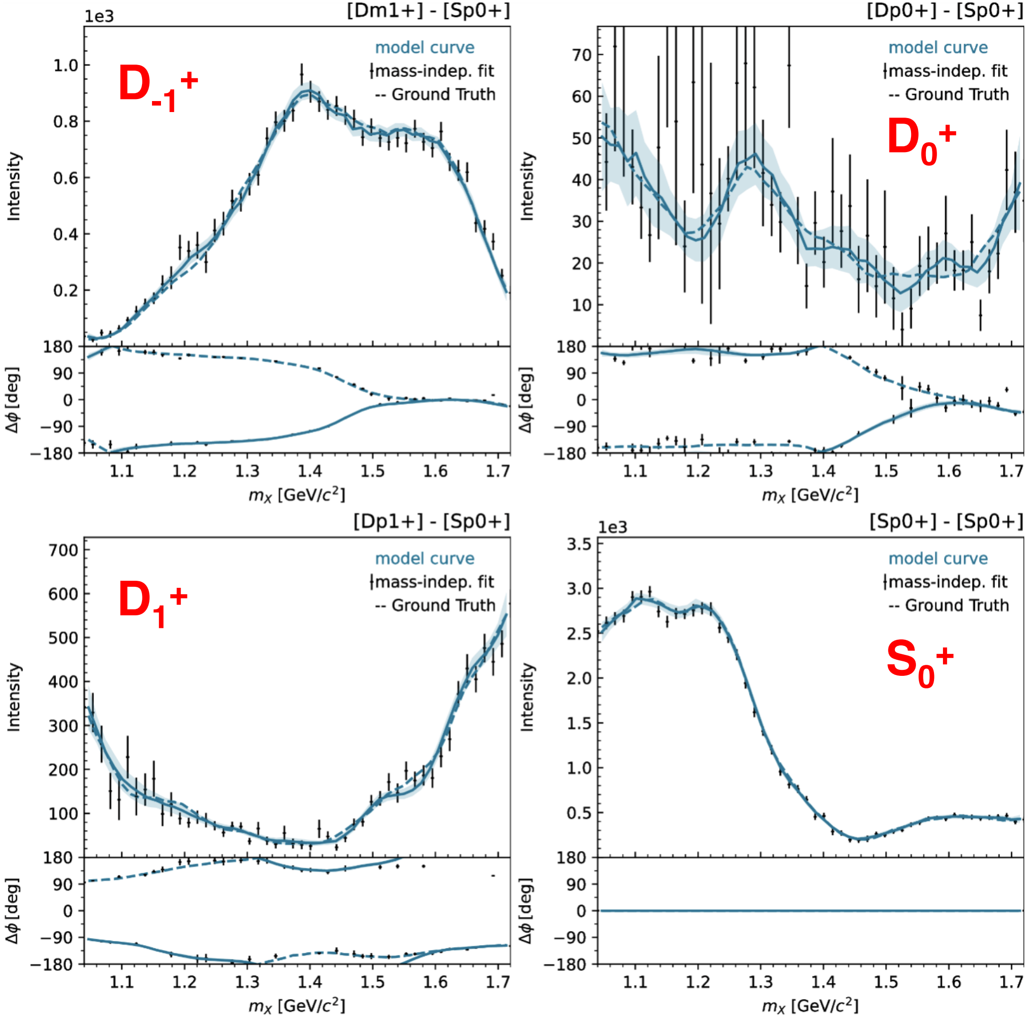
Positive reflectivity Waveset:
D_{-1}^{+} D_{0}^{+} D_{1}^{+} S_{0}^{+}

Single Prior Sample



Intensity

ΔPhase



Dashed blue line = ground truth
Blue line/fill = ift mean / stddev
Black error bars = Mass indep. fits

Both approaches perform well

Traditional binned fits have higher scatter

ift results:

- captures truth within uncertainties
- finds the trivial (phase-flip) ambiguity

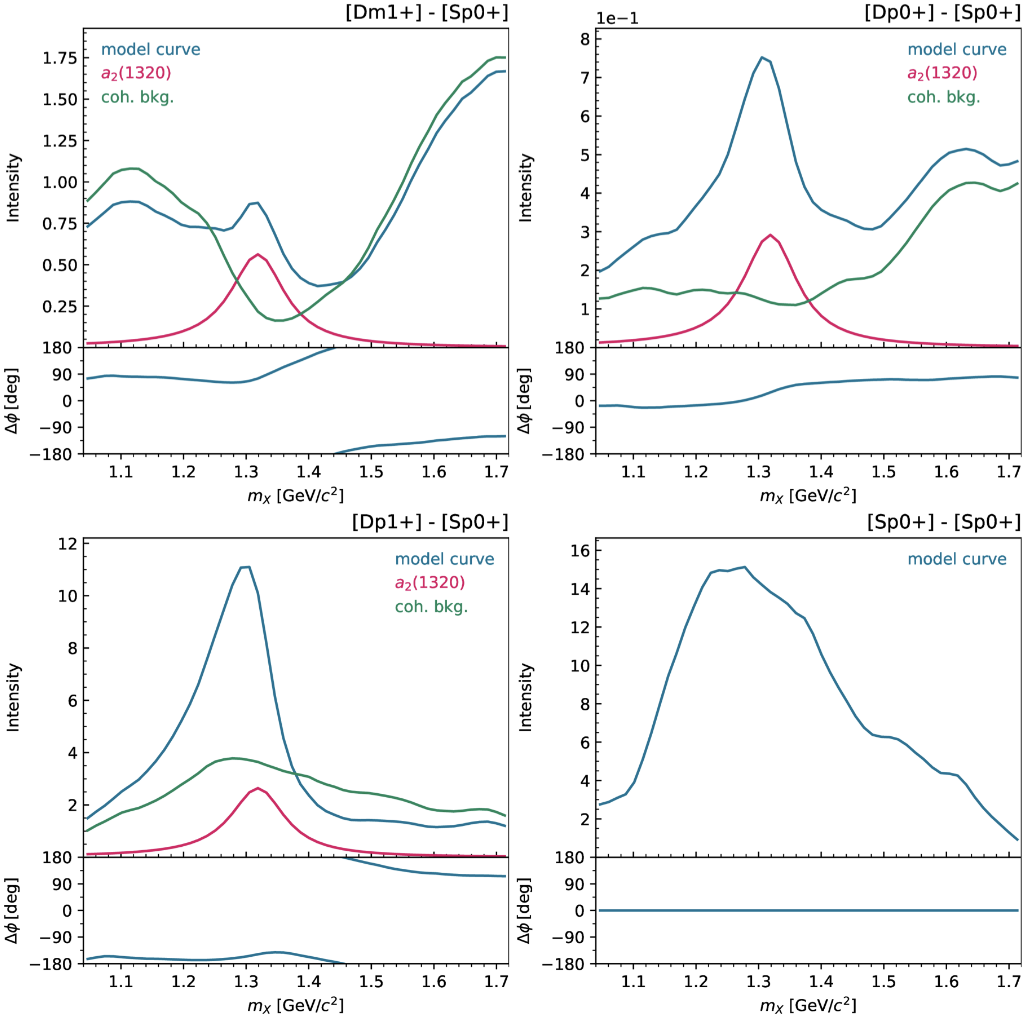
I/O Study 2

Same as Study 1 but with

a_2(1320) Breit-Wigner resonance

+

Coherent non-parametric background

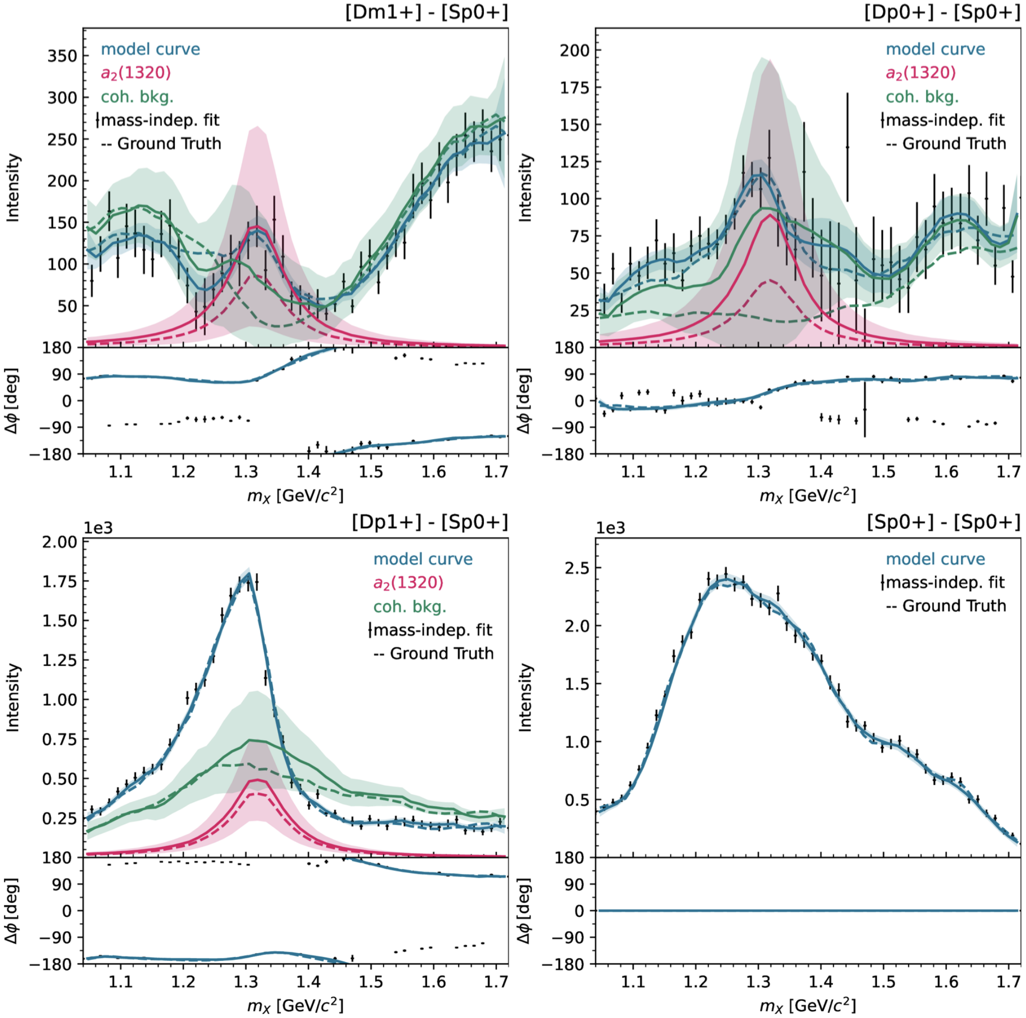


Single Prior Sample

model curve

$a_2(1320)$

coh. bkg.



model curve

$a_2(1320)$

coh. bkg.

Individual components are mostly recovered (within uncertainties)

Conclusion

- Partial wave analysis to determine spectrum of hadrons to study non-perturbative QCD
- `iftpwa` is a **complex-valued** model building framework allowing mixing of **parametric** and **non-parametric** components
- Upcoming publication on the method and release of the framework

Backup

Gaussian Process Prior

Power Spectrum

Gaussian Excitation Field

Position (amplitude space) offset

```
GP = HT( A * zero_mode * xi ) + offset
```

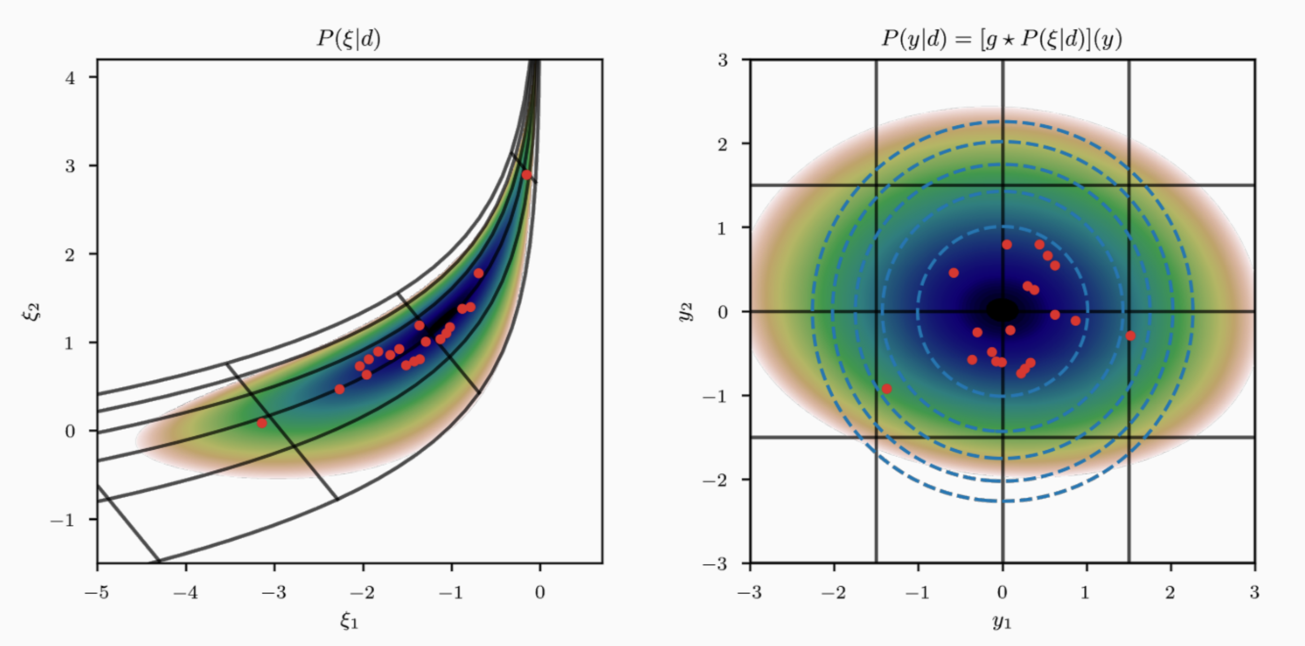
Harmonic Transform Frequency → Position (Amplitude) Space

Power Field Offset

Prior distributions defined by some set of hyperparameters

About 8 of them

Geometric Variational Inference (geoVI) [?]



[From Phillip Frank's backup slides on GeoVI]()

Information Hamiltonian $\mathcal{H}(\xi\|d)$:	$-\log\left(\mathcal{P}(\xi\|d)\right)$
Posterior metric $\mathcal{M}(\xi)$:	$\mathcal{M}_{\text{lh}}(\xi) + \mathbb{1}$
Fisher information metric $\mathcal{M}_{\text{lh}}(\xi)$:	$\left\langle \frac{\partial^2 \mathcal{H}(d\|\xi)}{\partial \xi \partial \xi'} \right\rangle_{\mathcal{P}(d\|\xi)}$

YAML Configuration

```yaml
GENERAL:
    pwa_manager: GLUEX
    ...

# Parameters of partial-wave manager class -> t
PWA_MANAGER:
    cfgfiles: ???
    min_mass: ???
    ...

IFT_MODEL:
    scale: 18000 # Intensity ~ 1e-4 for scale =
    positiveScale: true
    useLogNormalPriorSlope: false
    loglogavgslope: [-4.0, 1.0] # (mean, std)
    ...

LIKELIHOOD:
    approximation: false # no not approximate
    metric_type: normazl #[[2, 'normal'], [18,
    ...

# Parameters of optimization procedure
OPTIMIZATION:
    nSamples: 50 # [[5, 0], [10, 5], [15, 10],
    nIterGlobal: 22
    algoOptKL: LBFGS
    ...
```

ROOTPWA Public

PyAmpTools Public

Define GP Prior Model

Likelihood approx.

Optimizer specs

+

Parametric model Cfg

```python
def etapi_a2a2p():

    m_a2_1320 = LogNormal(sigma=0.0013 * 30, mean=1.3186)
    w_a2_1320 = LogNormal(sigma=0.002 * 30, mean=0.105)

    m_a2_1700 = LogNormal(sigma=0.05, mean=1.700)
    w_a2_1700 = LogNormal(sigma=0.05, mean=0.300)

    resonances = {
        "a2_1320": {
            "name": "$a_2(1320)$",
            "fun": breitwigner_normed,
            "paras": {"mass": m_a2_1320, "width": w_a2_1320},
            "waves": [
                'reaction_000::NegIm::Dm2-',
                'reaction_000::NegIm::Dm1-',
            ],
        },
        "a2_1700": {
            "name": "$a_2(1700)$",
            "fun": breitwigner_normed,
            "preScale": 0.25,
            "paras": {"mass": m_a2_1700, "width": w_a2_1700},
            "waves": [
                'reaction_000::NegIm::Dm2-',
                'reaction_000::NegIm::Dm1-',
            ],
        },
    }

    smoothScales = False
    return resonances, smoothScales
```

Resonance parameter priors

Resonance specs as a dictionary

Prior Model Specification and Hyperparameters

- Prior model: can have lots of hyperparameters
- Optuna: allows black-box hyperparameter optimization
  - handful of optimization algorithms (samplers)
- Define in YAML format the optimization criteria, sampler, search space

```yaml
HYPERPARAMETER_SCANS:
    n_trials: 10
    objective: "minimize|energy"
    sampler: RandomSampler # BruteForceSampler, null
    PARAMETRIC_MODEL.resonances.0.a0_980.preScale|suggest_float: "0.1, 5.1, step=0.5"
    PARAMETRIC_MODEL.resonances.2.a2_1700.preScale|suggest_float: "0.1, 1.0, step=0.1"
    IFT_MODEL.res2bkg|suggest_float: "0.1, 2.1, step=0.5"
```

Hyperparameter values at best objective not always the best!

OPTUNA