Image credit: Marguerite Tonjes

Systematics

Nick Smith

Future Trends in Nuclear Physics Computing

29 September, 2022

Outline

- What are systematics?
- Basic tasks in analysis
  - Modeling
  - Evaluation
  - Book-keeping

- Goals:
  - Review present thinking in
    - Technical design of systematics handling in analysis software and data flow
    - Approaches to systematics modeling and presentation
  - Encourage discussion
    - Good abstractions?
- My perspective:
  - CMS Higgs combination convener
  - Coffea analysis framework developer
  - Statistics hobbyist

What are systematic effects?

- Ingredients in defining how p depends on θ
  - And the data y
  - And the dimension of θ

(μ = parameters of interest)

Nuisance parameters are generally included in a model to take into account systematic uncertainties. Suppose that x are the *primary measurements* and have probability (density) $p(x|\mu,\theta)$. In order to constrain the nuisance parameters θ we have a set of independent *auxiliary data* y, with probability $p(y|\theta)$. The joint probability of the data x and y is, therefore, given by

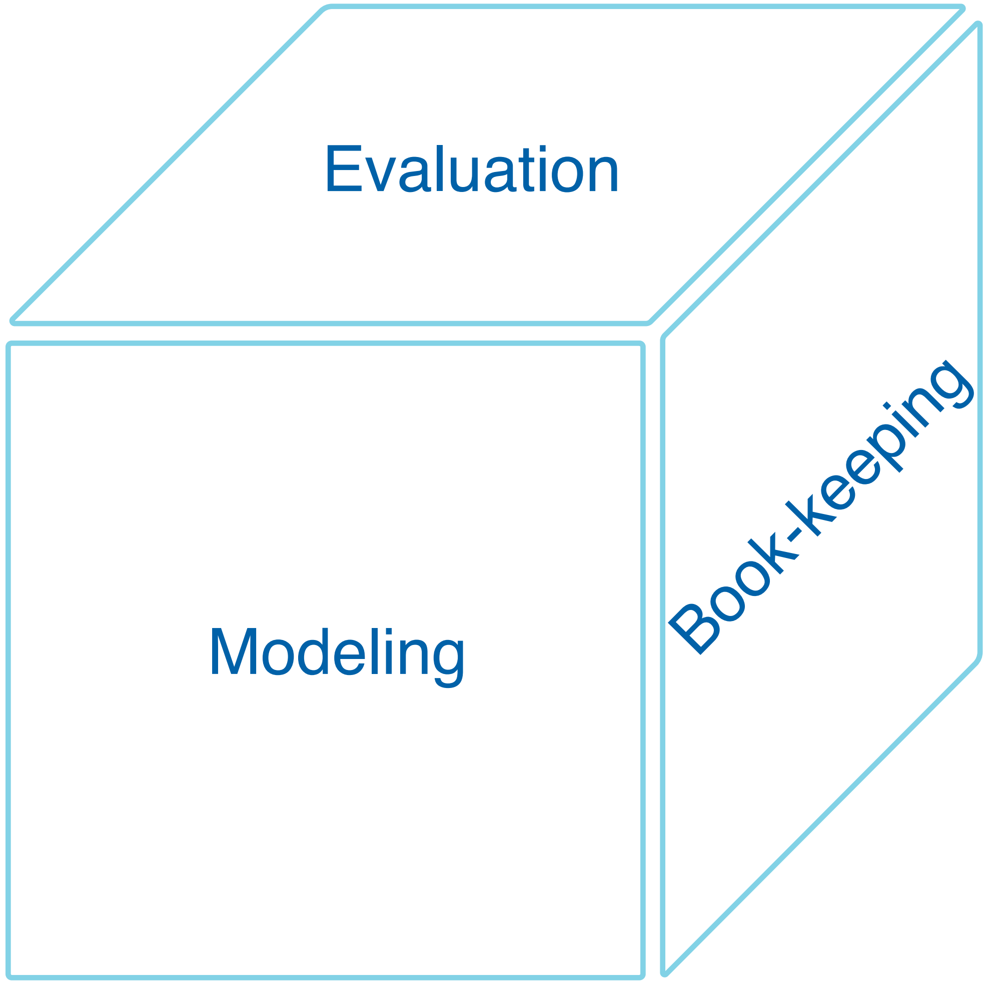
$$p(x,y|\mu,\theta) = p(x|\mu,\theta)p(y|\theta)\ . \qquad (2)$$

arxiv:2109.04981

- Frequentist interpretation: p is likelihood
  - Profile or marginalize away θ
  - Formalism allows Bayesian approach as well
    - Know any PyMC3/BAT.jl/etc. analyses?

Typical tasks

- Enumerate effects to get dimension of θ
  - Don't forget anything! Unknown unknowns?
- Choose a function class $p_\eta(y|\theta)$
  - Also for p(x)
- Evaluate η
  - In practice: interpolate between several $p(y|\theta_i)$
- Iterate
  - Compromise: fidelity/computability/practicality
    - Prune low-impact effects
  - Initial model might not fit observed y well
- We want to simplify these tasks



Disclaimer

- My perspective: CMS Higgs combination
  - Mostly binned likelihood models
    - Cannot evaluate density, but can sample
    - Some unbinned p(x) estimated from samples
- Common ground: Poisson process?
  - One *can* model with binned approach
  - Unbinned = limit of small bins

$$\int_{\mathrm{bin}} P(x)\mathrm{d}x \approx \frac{\sigma}{N} \sum_{x_i \sim P(x)}^{N} 1(x_i \in \mathrm{bin})$$

$$\cancel{P(x)}\mathrm{d}x$$

where else. Orear [1] makes plain what is happening by dividing the range of x into narrow bins of width Δx, so small that the probability of a bin containing more than one event is negligible. The probabilities for 0 and 1 events in a bin are given by Poisson statistics

$$P_0(x) = \mathrm{e}^{-\Delta x P(x)},$$

$$P_1(x) = \Delta x P(x)\, \mathrm{e}^{-\Delta x P(x)}.$$

The extended likelihood $\mathscr{L}$ is thus the combined probability for a complete data sample

$$\mathscr{L} = \prod_i \Delta x P(x_i) \prod_j \mathrm{e}^{-\Delta x P(x_j)},$$

where the first product is over all bins containing an event, and the second is over all bins.

In the limit $\Delta x \to \mathrm{d}x$, the first term becomes $\prod_i P(x_i)\, \mathrm{d}^N x$, analogously to eq. (1) – the $\mathrm{d}^N x$ merely expresses the fact that the functions are probability densities, not probabilities. The second is

$$\mathrm{e}^{-\Sigma_j \Delta x P(x_j)} \to \mathrm{e}^{-\int P(x)\mathrm{d}x} = \mathrm{e}^{-\mathscr{N}}.$$

Thus the extended likelihood is given by

$$\mathscr{L} = \left[\prod_i P(x_i)\right] \mathrm{e}^{-\mathscr{N}}. \qquad (5)$$

R. Barlow, Extended maximum likelihood
doi:10.1016/0168-9002(90)91334-8

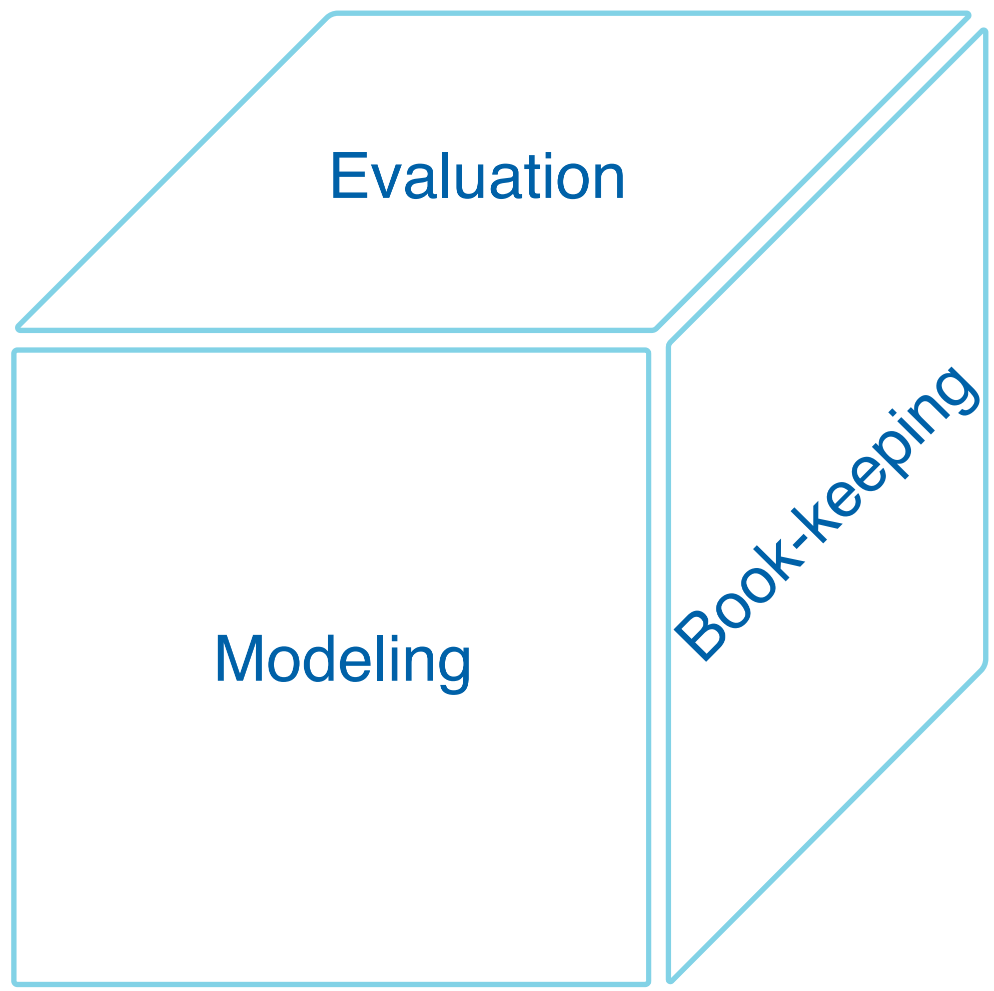
Relevant topics

Analysis
task graph

Reduced
data formats

On-the-fly
evaluation

Differentiable
analysis



Systematic-aware
optimization

Metadata
systems

Modeling
techniques

Experimental
prescriptions

Publishing
statistical models

Template
storage

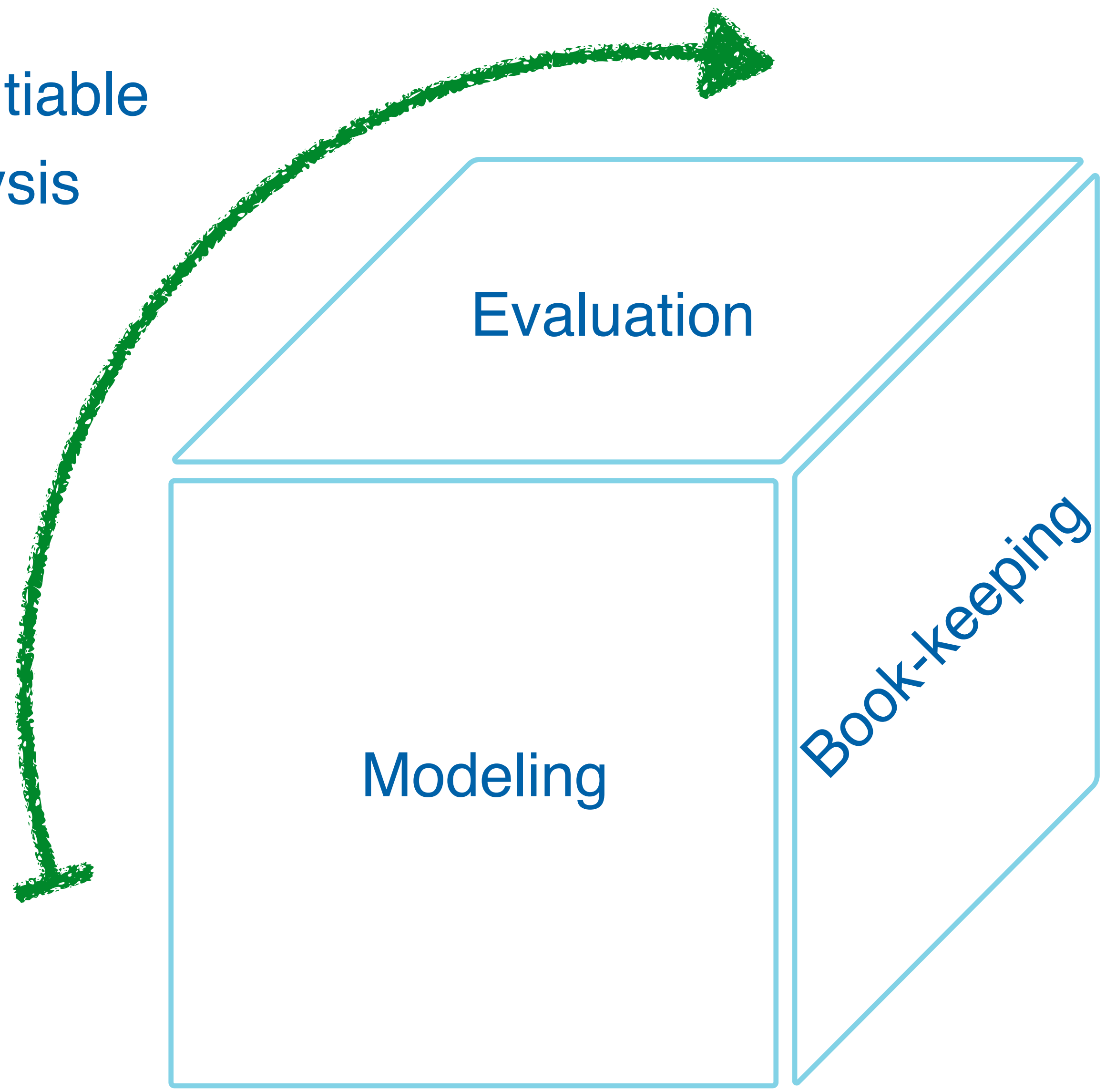
Relevant topics

Analysis
task graph

Reduced
data formats

On-the-fly
evaluation

Differentiable
analysis



Systematic-aware
optimization

Metadata
systems

Modeling
techniques

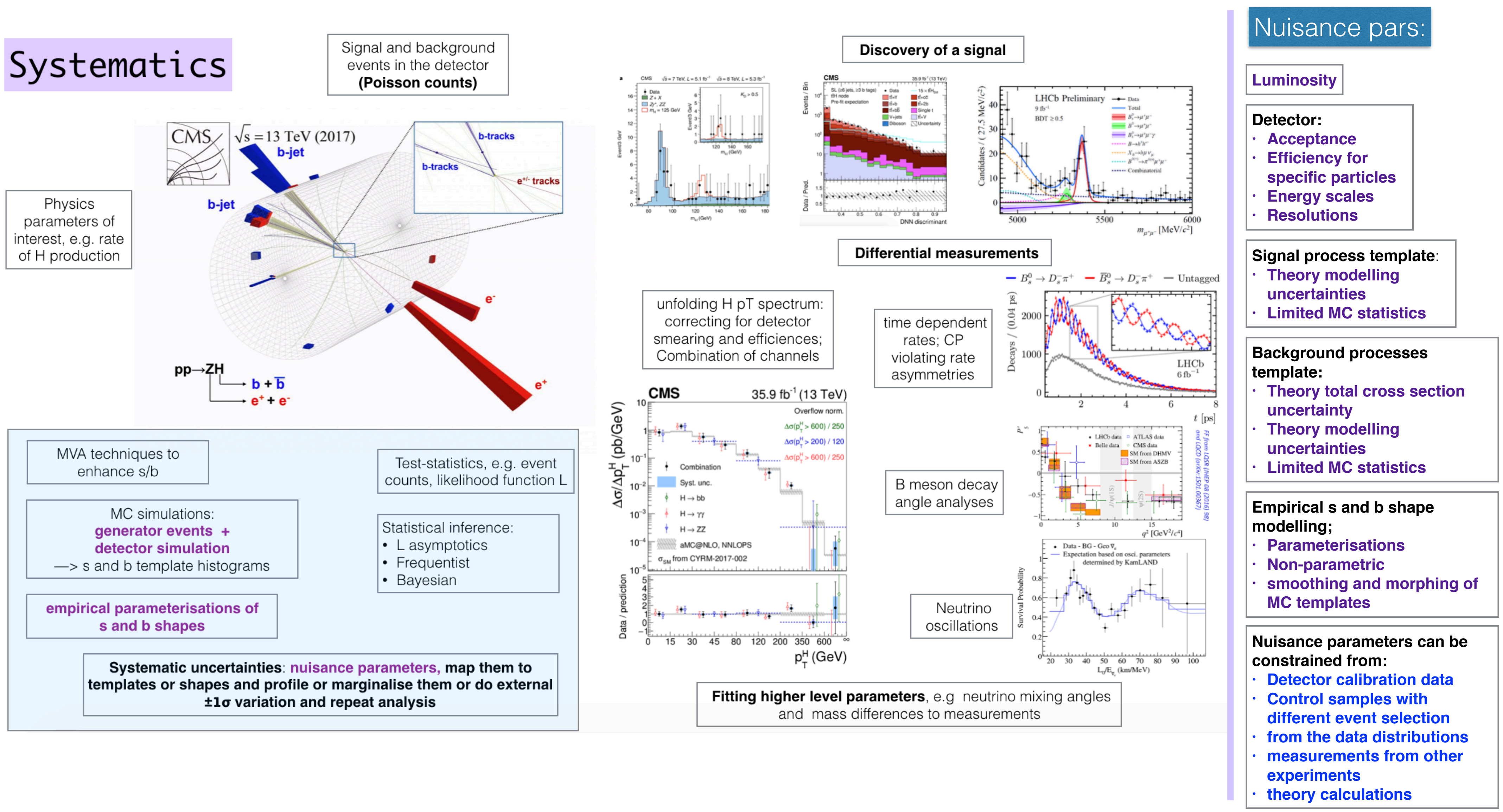
Experimental
prescriptions

Publishing
statistical models

Template
storage

Modeling techniques

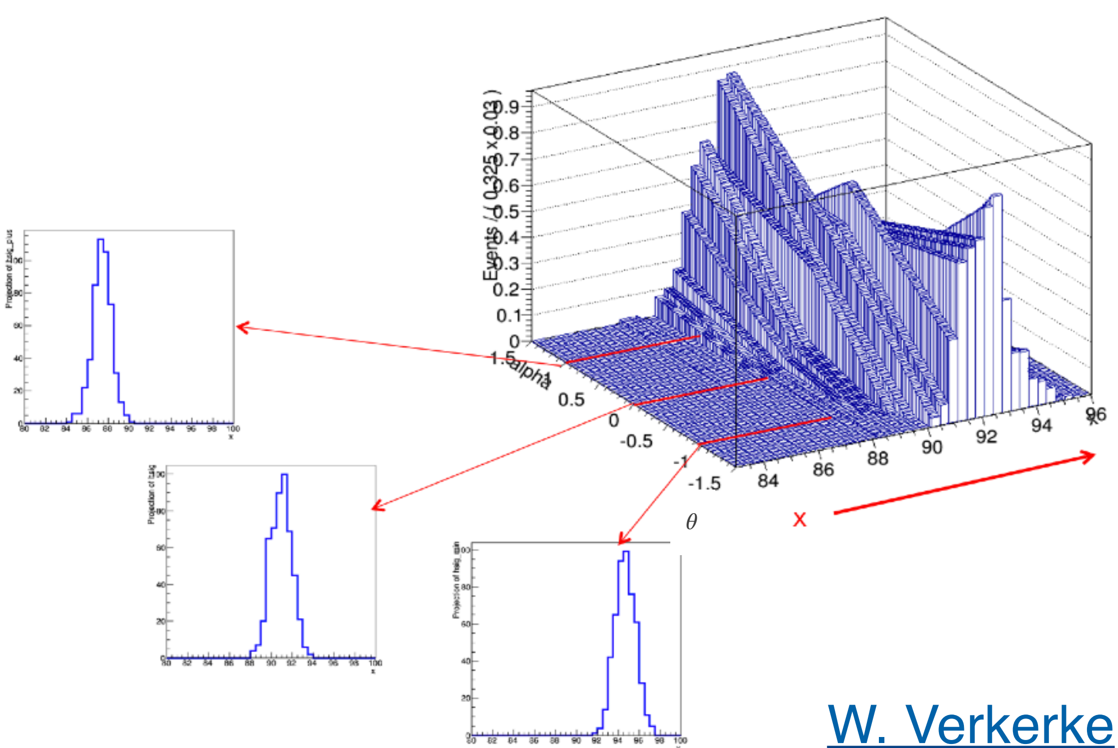
- Can be a whole workshop
  - Was: [PHYSTAT-Systematics 2021]
  - Excellent presentations covering a wide range of techniques
  - A one-slide overview was [produced] (more on types than techniques)



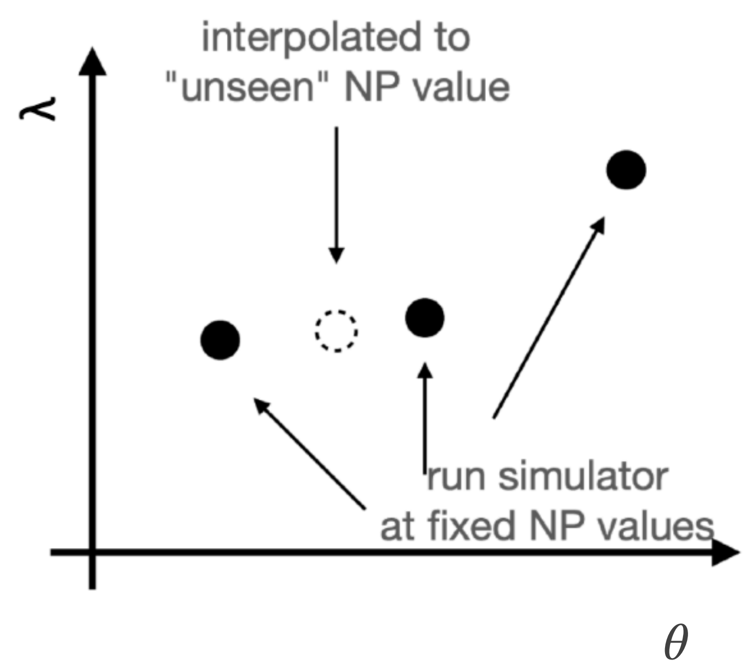
Modeling techniques

- **Rich** set of interpolation/extrapolation techniques at end-stage
  - Morphing: vertical, horizontal, moment; splines; gaussian process; asymmetric shift interpolation; additive/multiplicative effects; MC stat uncertainty, BB-lite; …
  - i.e. what is done in RooFit/pyhf/zfit/iMinuit/combine/etc.
    - What features do each of these tools offer? Nobody has it all!

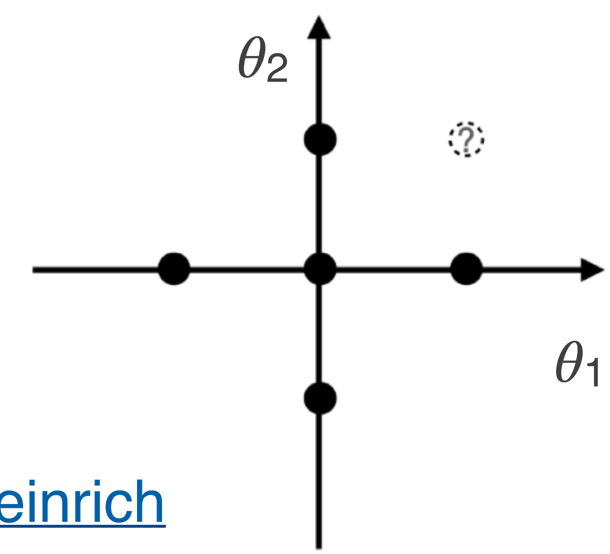
Visualization of bin-by-bin linear interpolation of distribution



W. Verkerke



Combining effects



L. Heinrich

Modeling techniques

- Simpler taxonomy of techniques to get inputs to fitting tools?
  - This is the dominant analysis-stage computation expense (process billions of events)
- Posit three basic techniques
  - I think all of these can be done unbinned as well
  - Just need functions w(x,θ) and Δ(x,θ)

$$\int_{\mathrm{bin}} P(x)\mathrm{d}x \approx \frac{\sigma}{N} \sum_{x_i \sim P(x)}^{N} 1(x_i \in \mathrm{bin})$$

(nominal)

$$\int_{\mathrm{bin}} P(x|\theta=\theta_1)\mathrm{d}x \approx \frac{\sigma}{N} \sum_{x_i \sim P(x|\theta=\theta_1)}^{N} 1(x_i \in \mathrm{bin})$$

(alternative sample, e.g. 2-point)

$$\int_{\mathrm{bin}} P(x|\theta=\theta_1)\mathrm{d}x \approx \frac{\sigma}{N} \sum_{x_i \sim P(x)}^{N} w(x_i, \theta=\theta_1) 1(x_i \in \mathrm{bin})$$

(reweight, e.g. efficiency)

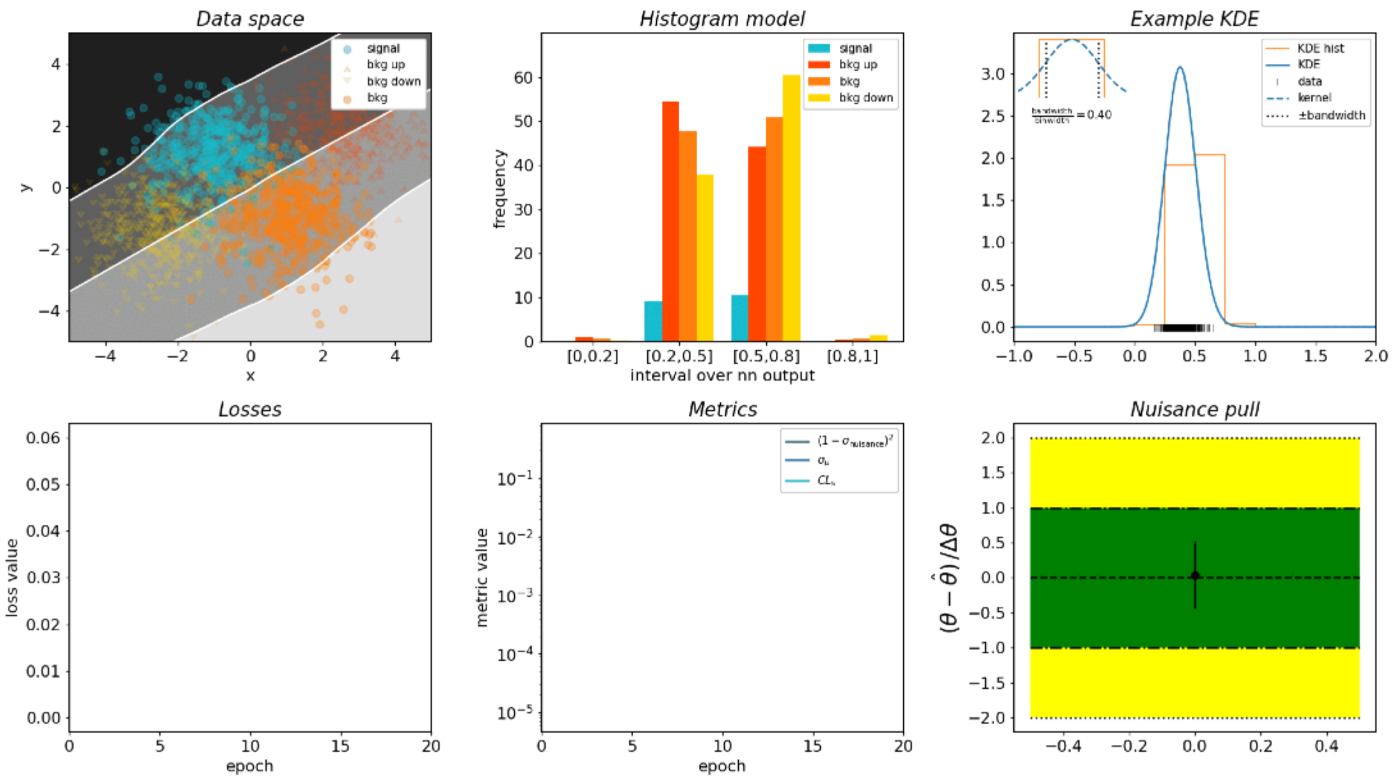
$$\int_{\mathrm{bin}} P(x|\theta=\theta_1)\mathrm{d}x \approx \frac{\sigma}{N} \sum_{x_i \sim P(x)}^{N} 1(x_i + \Delta(x_i, \theta=\theta_1) \in \mathrm{bin})$$

(shift, e.g. energy scale)

Systematic-aware optimization

- Analysis design and optimization often involves ML these days
- Learn salient features, ignore features affected by nuisance params
- Dozens of proposals, see HEPML LivingReview sections:
  - Decorrelation methods allow for construction of control regions
  - Inference-aware: maximize sensitivity or exclusion power of POI in full likelihood model
    - Can be deployed in more “traditional” analyses for e.g. region/binning optimization
  - Domain adaptation: ensure marginalized observables are modeled well

neos: N. Simpson, L. Heinrich



Fermilab

Differentiable analysis

- Rather than θ up/down variation, compute value and gradient
  - Auto-diff vs. finite-diff performance
- Higher order derivatives? How analytic are these things?
  - Need second order to get asymmetric (and it probably does not extrapolate well :)
- Lukas has said more

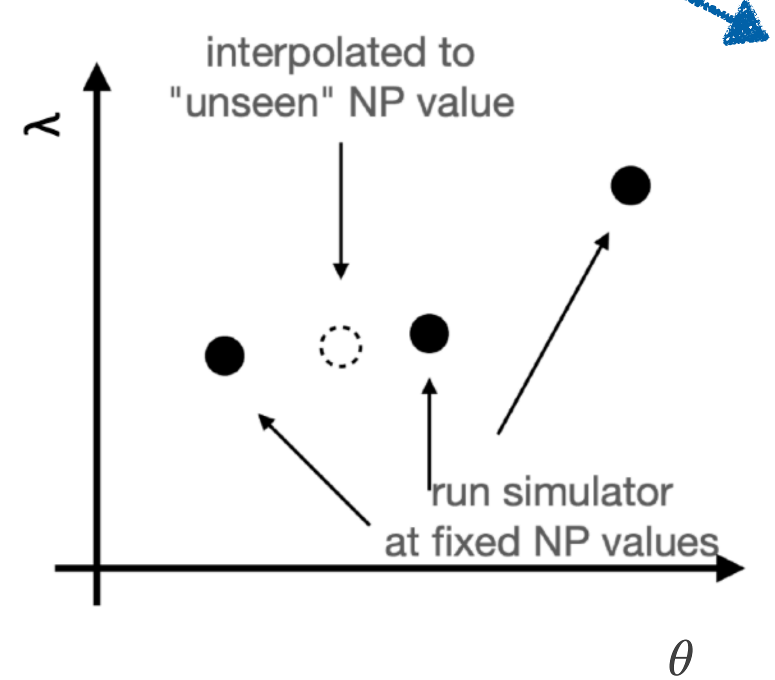
$$\lambda(\theta) = \frac{\sigma}{N} \sum_{x_i \sim P(x)}^{N} w(x_i, \theta) 1(x_i \in \text{bin})$$
$$\approx \lambda(\theta_0) + \frac{d\lambda}{d\theta}\Big|_{\theta=\theta_0} (\theta - \theta_0) + \cdots$$

(reweight, e.g. efficiency) 👍

$$\lambda(\theta) = \frac{\sigma}{N} \sum_{x_i \sim P(x)}^{N} 1(x_i + \Delta(x_i, \theta) \in \text{bin})$$

(shift, e.g. energy scale) 🤔

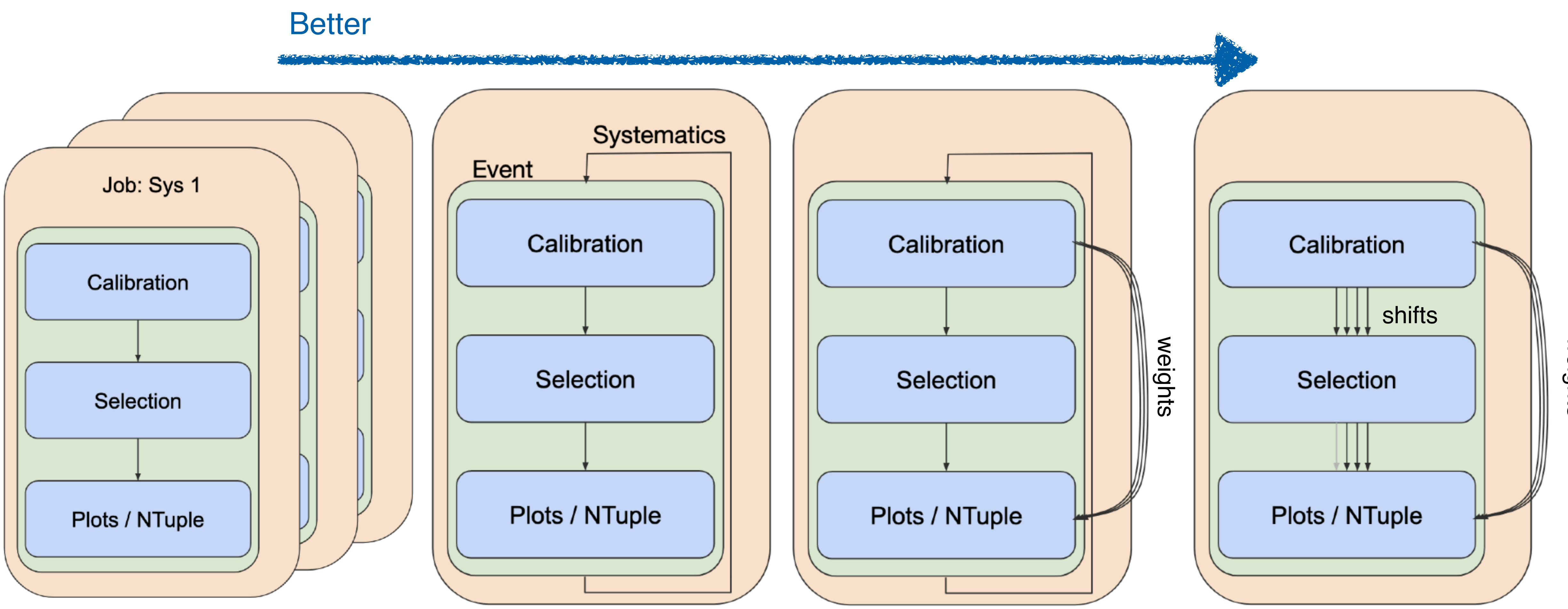
What happens here?



Fermilab

Analysis task graph

- Simplest solution: re-run everything with alternate θ
- Better: loop over event while in-memory (likely CPU cache)
  - Why? Because IO is very expensive
- Best: compute all weights, compute shifts only as necessary



S. Hageboeck

Analysis task graph

- RDF Vary simplifies implementation!
  - Coffea as well once awkward v2 is out (to have full DAG)
    - Current: embed lazy-evaluated systematics in NanoEvents

Each __getattr__ initiates evaluation

`events.jets.JES_jes.up.pt`

In ROOT 6.26 (experimental)

Python

attach an up/down variation to "pt"

proceed as usual, as if working with nominal values only

obtain all variations

```python
nominal_hx =
  df.Vary("pt", "RVecD{pt*0.9, pt*1.1}", ["down", "up"])
   .Filter("pt > k")
   .Define("x", someFunc, ["pt"])
   .Histo1D("x")

hx = ROOT.RDF.VariationsFor(nominal_hx)
hx["nominal"].Draw()
hx["pt:down"].Draw("SAME")
```

E. Guiraud

Fermilab

Reduced data formats

- Goal: maximize usability, minimize disk space
  - Keep minimal subset of observables x
- Tradeoff with functions $w(x,\theta)$ and $\Delta(x,\theta)$:
  - Large subset of x needed to evaluate: better to save output for θ_0, θ_1, θ_2
  - Small subset of x needed to evaluate: better to save those inputs, evaluate “on-the-fly”
  - Overlap with what is needed to identify the bin ➜ more likely on-the-fly
- CMS NanoAOD: calibrated objects, very few systematics
  - Keep only those too difficult to parameterize
    - Unclustered energy Δ for MET: per-PF candidate species energy scale uncertainty
  - ATLAS DAOD_PhysLite: similar goals
- Other considerations
  - CMS MiniAOD: lossy compression of track covariance matrices
  - Common weight trick: store 1-w with reduced-precision mantissa

On-the-fly evaluation

- Often calibrations and systematics go hand-in-hand
  - Can redefine $p(y'|\theta') = p(y+\Delta(y,\theta_0)|\theta-\theta_0)$ so auxiliary measurement is "spot-on"
- In CMS, corrections+uncertainty have long been parameterized
  - Lately, move towards standardizing to reduce proliferation of (often poorly-designed) serialization formats and (often slow) evaluation frameworks
- Correctionlib
  - A well-structured JSON data format for a wide variety of ad-hoc correction factors encountered in a typical HEP analysis and a companion evaluation tool suitable for use in C++ and python programs.
  - Development started Nov. 2020, all CMS analysis-stage corrections now compatible
  - Presented at PyHEP '22: youtube

Python signature

```python
def f(*args: str | int | float) -> float:
    return ...
```

C++ signature

```cpp
double Correction::evaluate(const std::vector<std::variant<int, double, std::string>>& values) const;
```

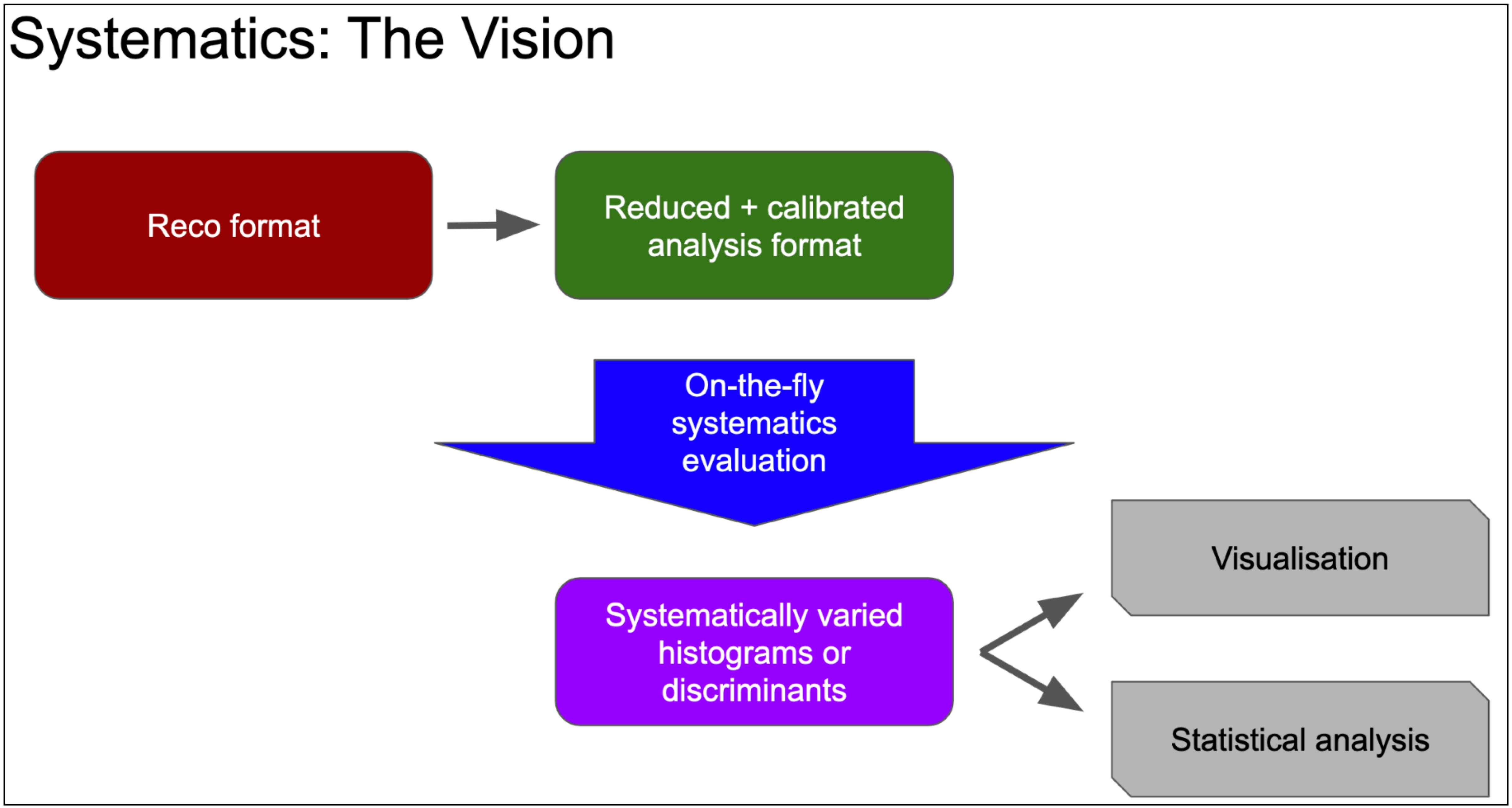
Metadata systems

$$\int_{\text{bin}} P(x|\theta=\theta_1)\mathrm{d}x \approx \frac{\sigma}{N} \sum_{x_i \sim P(x|\theta=\theta_1)}^{N} 1(x_i \in \text{bin})$$

(alternative sample, e.g. 2-point)

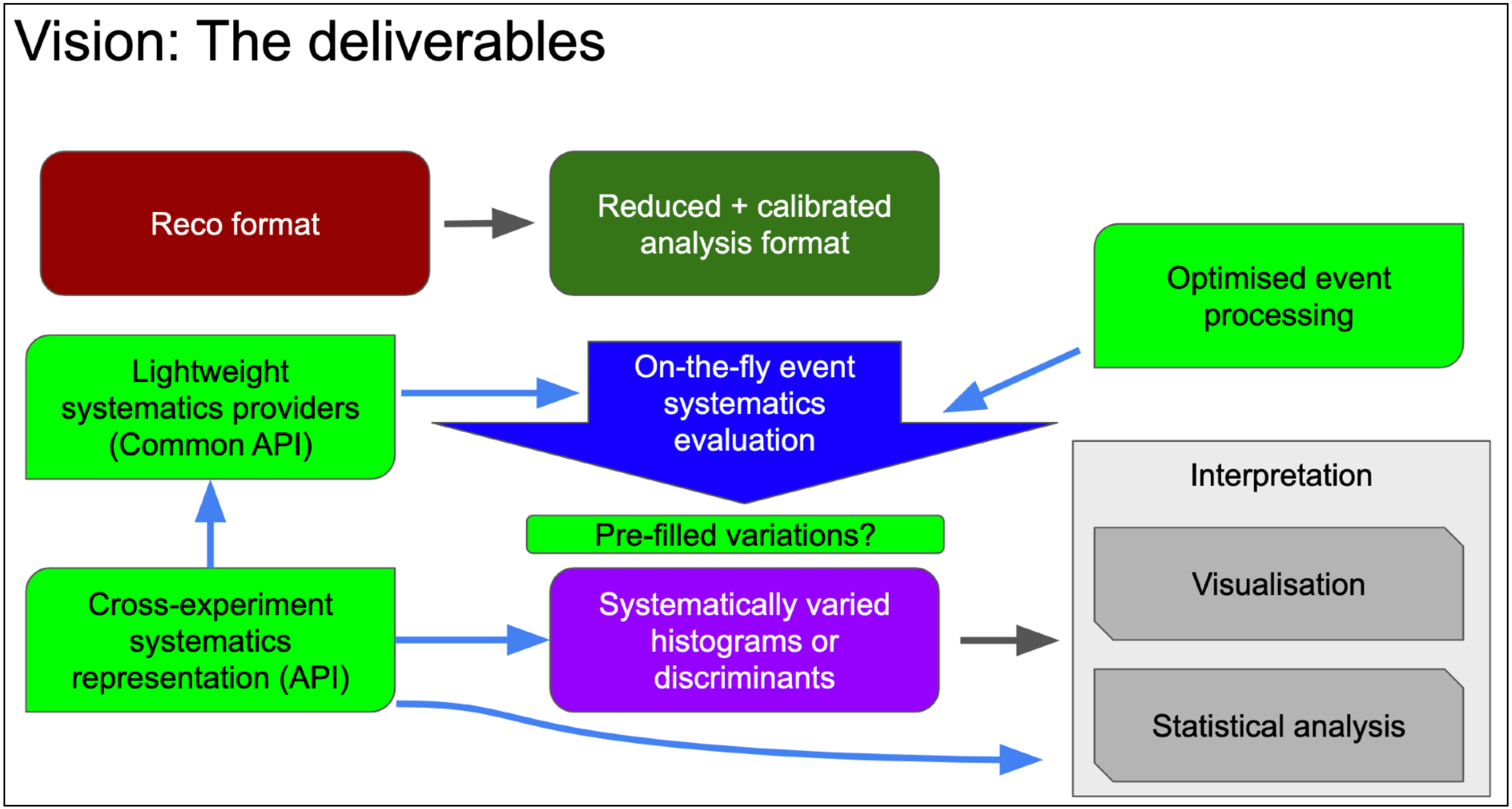
- Correctionlib json in database?
- Book-keeping alternative samples
  - At least in CMS, no automated access to generation config at analysis stage
  - Most book-keeping by hand: key on dataset name
- Paul has said more

A flowchart



P. Laycock, T.J. Khoo

A flowchart



[P. Laycock, T.J. Khoo]

A challenge

The “ttbar” systematics challenge

- Proposal for a DAWG follow-up
- Can ATLAS/CMS/LHCb do a joint (OpenMC) ttbar analysis?
  - Consistent MC events through collaboration reco
  - Prototype a common systematics representation
  - Propagate to (reasonable subset of) collaboration-specific systematics
  - Statistical combination of unfolded measurements
- Stretch goal: common cross-sections extracted from shared metadata API
- Show this off for AE3!?

P. Laycock, T.J. Khoo

Experimental prescriptions

- Non-trivial to agree on parameterization, but crucial for combinations
- Correlate (i.e. use same subset of θ for) common effects
  - Experimental effects (simplest: luminosity unc.)
  - Theory uncertainties for common processes
  - Etc.
  - Profit from increased sensitivity!
- CMS Higgs group: "datacards" (likelihood serialization format) are reviewed
  - Standard nuisances, naming conventions, sign, etc.
  - Simplifies combination later
- Is it worth establishing cross-experiment parameterization/nomenclature?
  - Test case: ttbar challenge?

Template storage

- Multi-dimensional histograms: axis for systematic variation
- Filling histograms with weights vector
  - Save repeated bin lookup for same observables
  - Planned feature for boost::histogram boostorg/histogram#211
- Better to have serializable object tailored to our use case
  - RDF has part of the answer:

```
hx = ROOT.RDF.VariationsFor(nominal_hx)
hx["nominal"].Draw()            obtain all variations
hx["pt:down"].Draw("SAME")
```

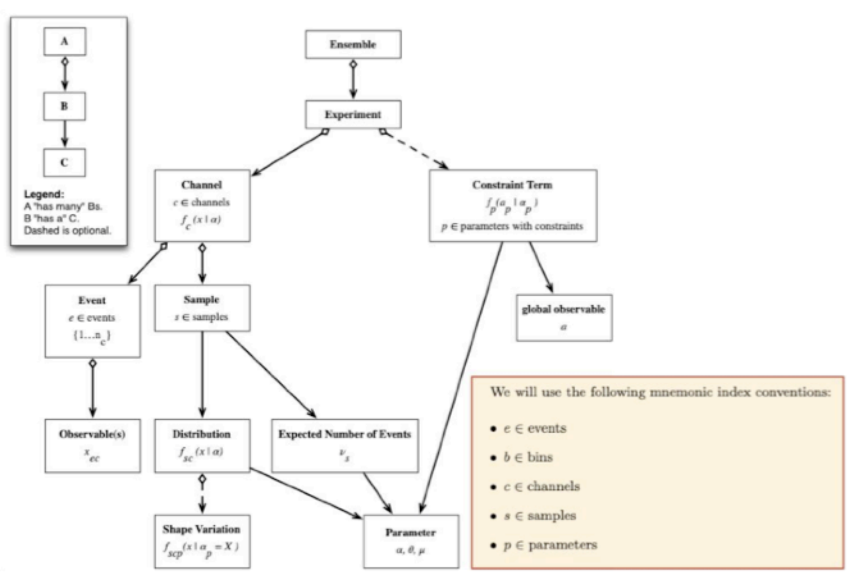
Template storage

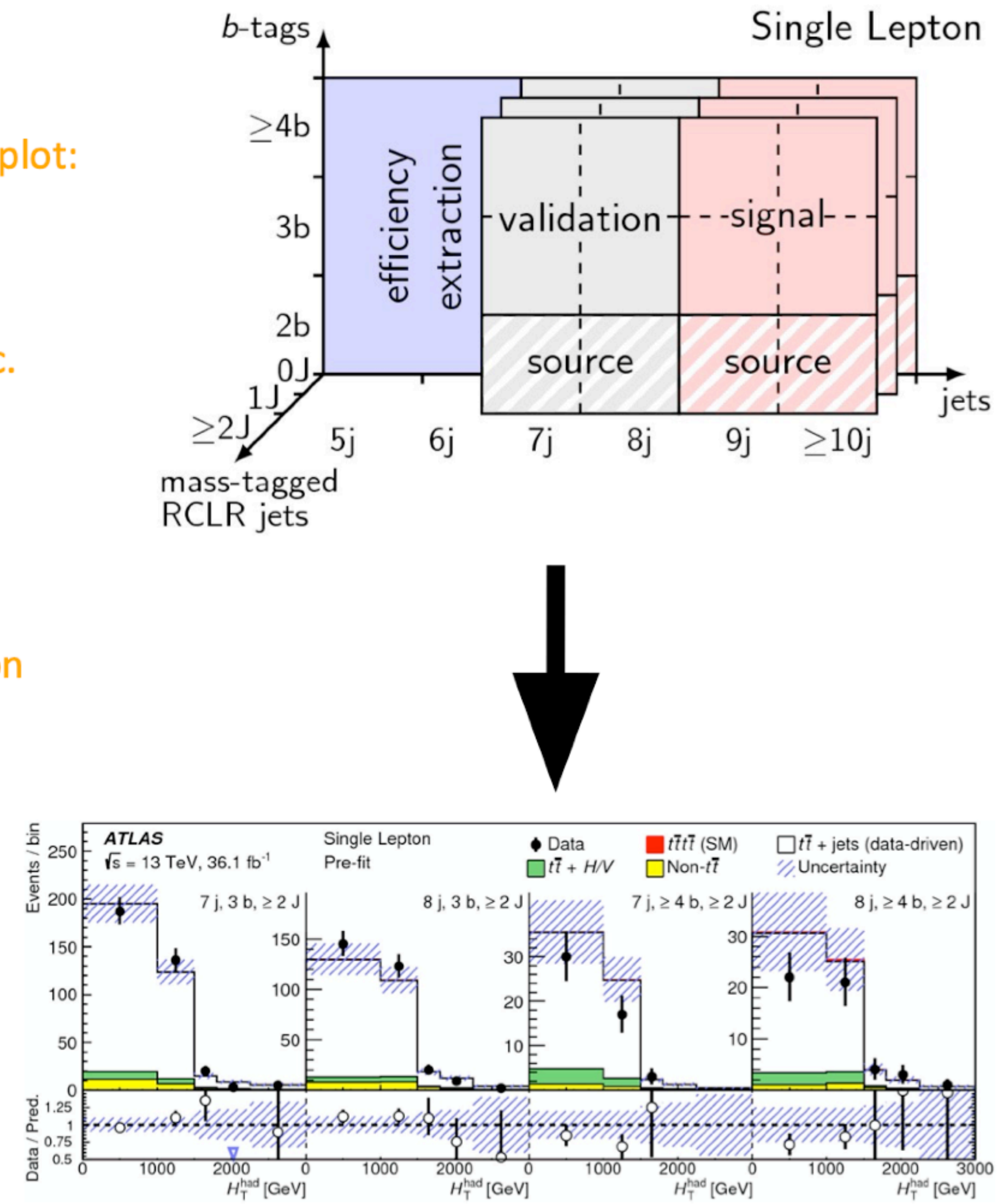
- cabinetry is a Python package to build and steer template fits

A point of convergence

Several aspects of Analysis Systems converge in a typical physics plot:

- Specification of signal / validation / control regions
- Specification of variables to be used for stat analysis
- Reduction to that format running on data and MC
- Management of MC samples, data driven backgrounds, etc.
- Management of systematic variations
- Feed reduced data (eg. histograms) into specification for statistical model / likelihood function
- Fitting & statistical tools
- Publishing results & derived data products
- Analysis preservation & gateways targeting reinterpretation





A. Held

Fermilab

Publishing statistical models

- More specifically the model $p(x, y|\mu, \theta)$
- Why?
  - "The statistical models used to derive the results of experimental analyses are of incredible scientific value and are essential information for analysis preservation and reuse … [and] can enhance the short- and long-term impact of experimental results." (arxiv:2109.04981)
- A goal now 22 years old (K. Cranmer)
- Need a good data format, contenders:
  - pyhf JSON (HistFactory XML)
  - CMS combine datacard
  - RooWorkspace json: HS3
    - "A round-trip-capable, human-readable declarative format for statistical models was missing"

HS3 specification

- Importers, exporters defined for many RooFit classes
  - User extendable
- Allows:
  - Easier conversion to other tools' formats
  - Faster/easier navigation in python code
  - Hand-editing
  - Simpler publication
- Many thanks to Carsten for starting this!

Transcript of a RooArgusBG example, arbitrarily selected analytical example

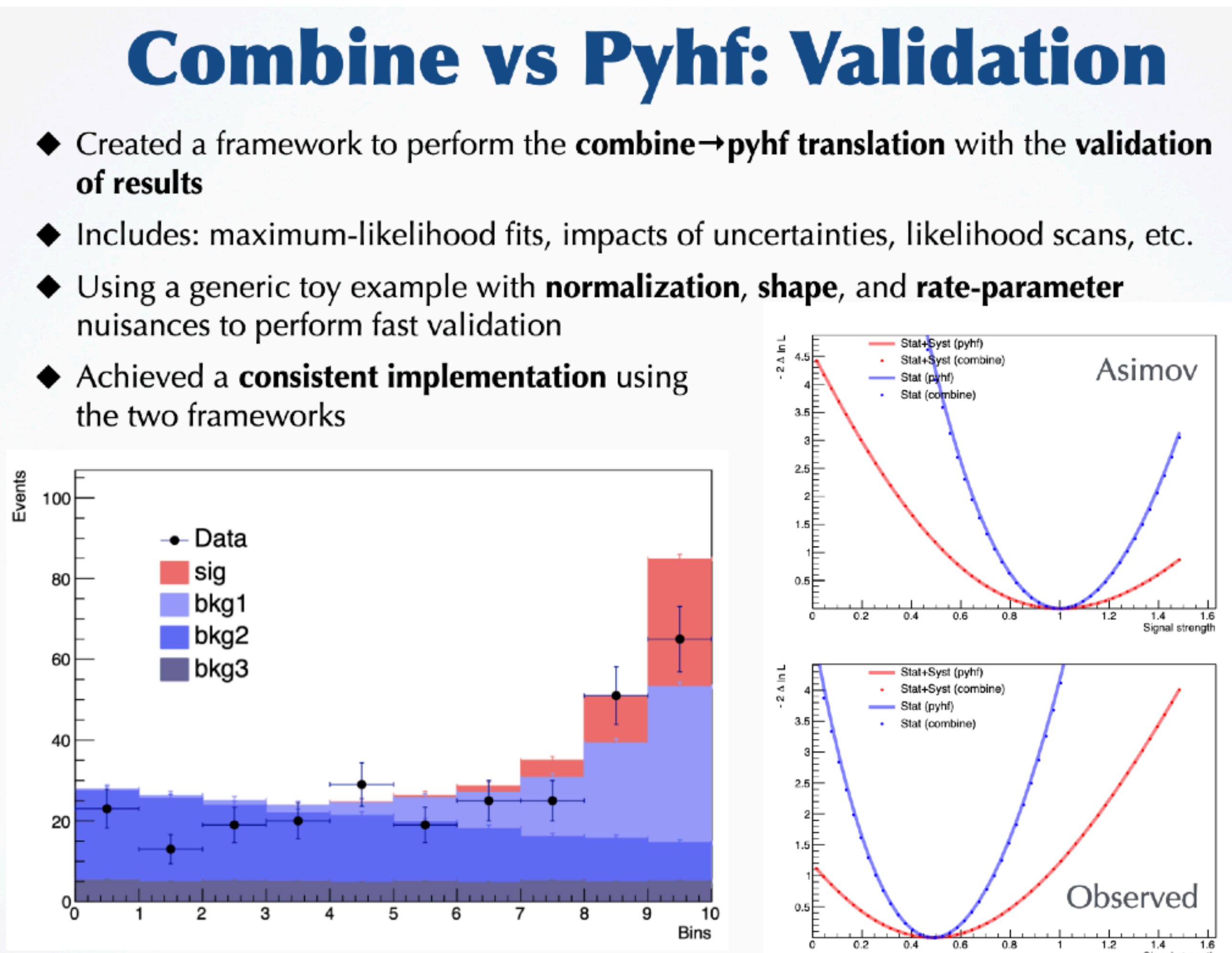
```
"pdfs": {
    "background": {
        "mass": "mes",
        "power": "0.5",
        "resonance": "5.291",
        "slope": "argpar",
        "type": "ARGUS"
    },
    "model": {
        "coefficients": [
            "nsig",
            "nbkg"
        ],
        "dict": {
            "ModelConfig": "ModelConfig"
        },
        "summands": [
            "signal",
            "background"
        ],
        "tags": [
            "toplevel"
        ],
        "type": "pdfsum"
    },
    "signal": {
        "mean": "sigmean",
        "sigma": "sigwidth",
        "type": "Gaussian",
        "x": "mes"
    }
},
```

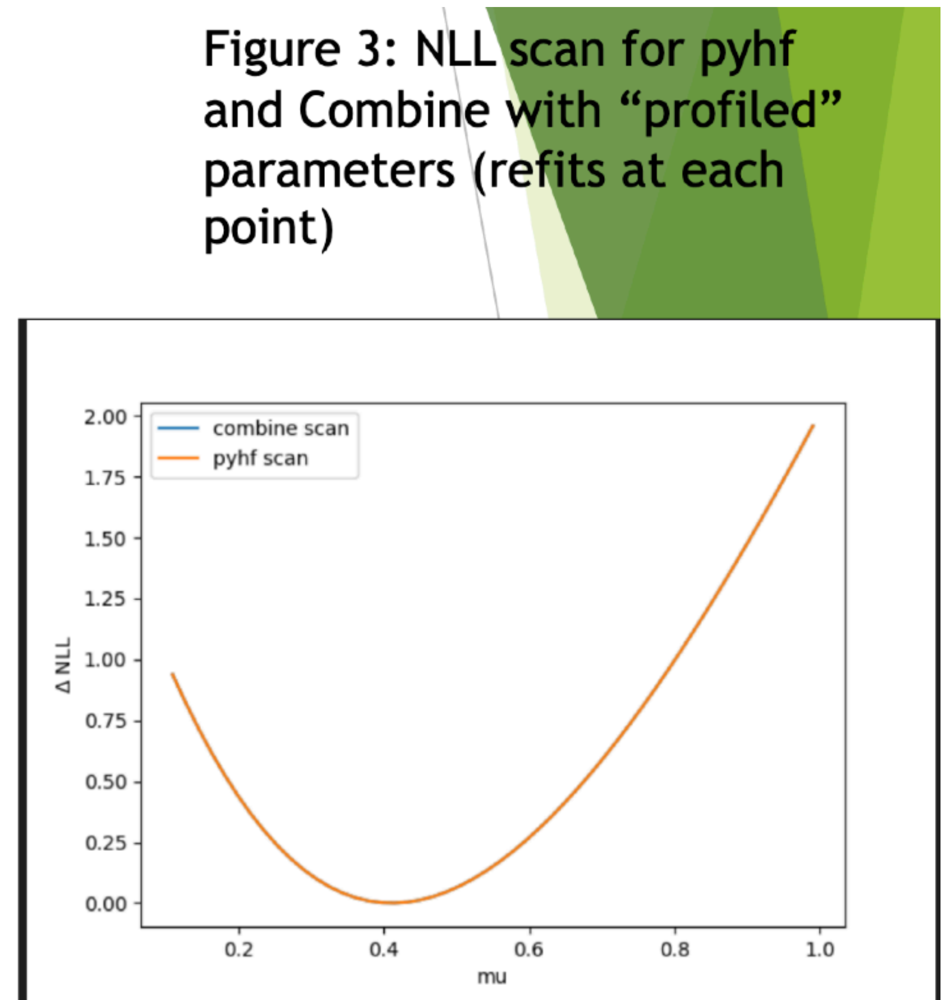
```
"variables": {
    "argpar": {
        "max": -1.0,
        "min": -100.0,
        "value": -20.0
    },
    "mes": {
        "max": 5.3,
        "min": 5.2,
        "value": 5.25
    },
    "nbkg": {
        "max": 10000.0,
        "min": 0.0,
        "value": 800.0
    },
    "nsig": {
        "max": 10000.0,
        "min": 0.0,
        "value": 200.0
    },
    "sigmean": {
        "max": 5.3,
        "min": 5.2,
        "value": 5.28
    },
    "sigwidth": {
        "max": 1.0,
        "min": 0.001,
        "value": 0.0027
    }
}
```

C. Burgard

Combine⇔pyhf

- Meanwhile, active work to compare CMS combine and pyhf tools
  - In the end, differences in terminology and matters of taste





[K. Skopven](#) [P. Ridolfi](#)

Summary

- Systematics influence **many** technical decision in analysis software
- Modern approaches to systematics handling are simplifying our life
  - More diversity in analyses will help us design better abstractions
- There is an active and enthusiastic community

- Further references:
  - Systematics for data analysis
  - AE2 workshop: systematics, summary
  - PHYSTAT-Systematics 2021
  - Publication of statistical models: hands-on workshop
  - Coffea+RDF discussion on systematics