Uncertainties and Unfolding:

A biased and incomplete picture

France-Berkeley PHYSAT Meeting on Unfolding

K. Cormier

Types of Uncertainties (for the purposes of this talk)

- Standard: Detector Systematics and Statistical Uncertainties

- Unfolding Specific:
  - Regularization Uncertainties, Unfolding Model Uncertainties

- New methods: incorporating uncertainties

‘Standard Uncertainties’: Detector Systematics and Statistical Uncertainties

“The Usual” – Detector and Statistical Uncertainties

- Present in every analysis
  - In principle: nothing special with respect to unfolding
- In practice: tools often shape treatment

Most common non-unfolding analyses:
Maximum Likelihood Estimation using
Profile Likelihood Uncertainties

Almost never seen (by me) outside of unfolding:
“one-at-a-time” uncertainties
sum uncertainty sources in quadrature

"One-at-a-time"

- Assumes no interaction between sources (linearity)
  - i.e. that not only *sources* of uncertainty are independent, but also *effects*
    Sounds very reasonable under the 'toy' folding/unfolding picture: $\vec{y} = \mathbf{M}\,\vec{x}$

But … this is linear in **M,** not in $\mathbf{M}^{-1}$

- this assumption also necessarily broken by a number of factors:
  - Normalized differential cross sections
  - Non-negative bin counts
  - Regularization – (to be discussed more later)

Unfolding Statistical Uncertainties

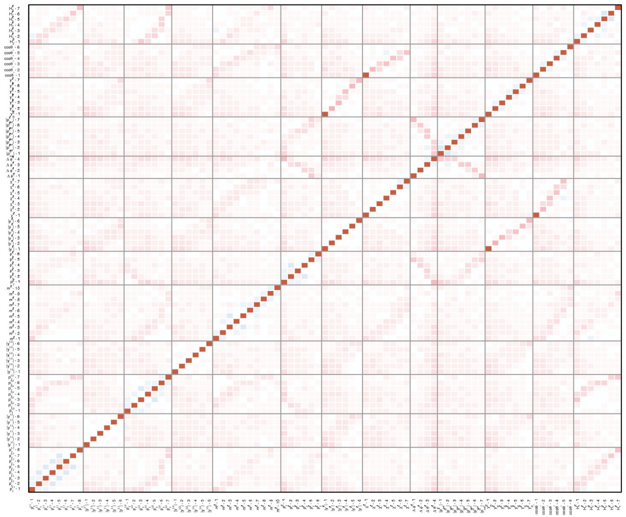
Can be estimated using Poisson toys:

1. For every binned count:
    sample from a poisson with that mean
2. Unfold

Repeat 1+2 many times, calculate covariances on the ensemble

Or: Estimate **correlations between statistically correlated observables** using the same data sample by resampling individual events

Statistical correlation matrix for 13 observables measured with the same dataset



https://arxiv.org/abs/1801.02052

One-at-a-time: Covariance construction + Uncertainty breakdown

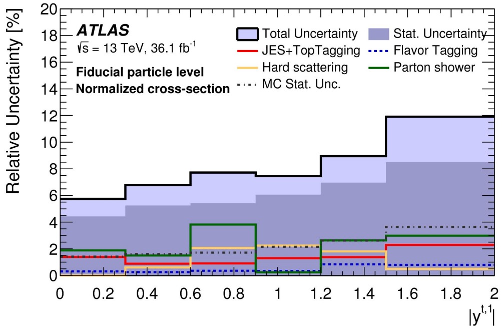
Covariance construction is simple:

$$Cov_{tot} = Cov_a + Cov_b + \ldots + Cov_n$$

But the cost is the assumption of linearity

Uncertainty breakdown is essentially free (but this is by assumption)

Breakdown of uncertainty sources can facilitate combinations



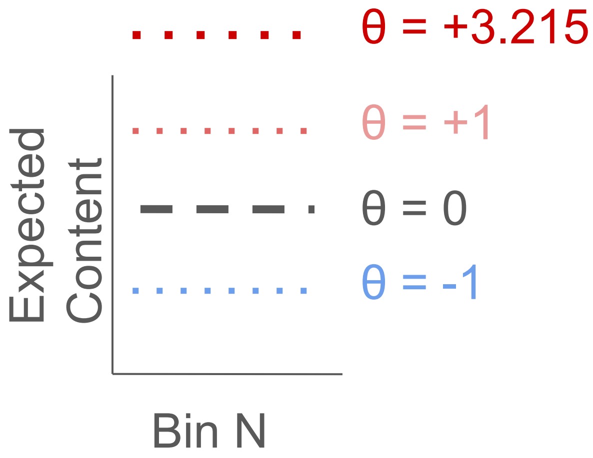
Toys for systematics

Can use **e.g. frequentist toy paradigm** to move beyond the 'one-at-a-time' methods

1. Start with nominal + systematic templates
2. Define a smooth scaling between them as a function of 'nuisance parameter' θ
3. Assign θ some pdf (often gaussian)

Repeat 1-3 for all uncertainties

4. Sample from $\vec{\theta}$, and derive toy bin content
5. Unfold



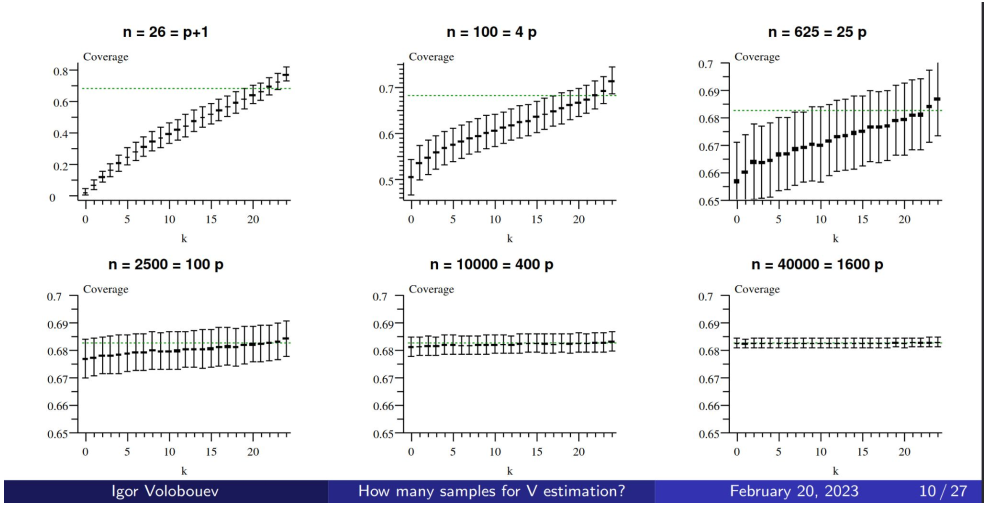
Repeat 4 & 5 N times, use ensemble to construct variances + covariances

A word of caution

Problem Setup

- We would like to estimate the covariance matrix for some measurement of a multivariate quantity, perhaps in an unfolding scenario.
- The distribution of results can not be derived analytically, but can be sampled from. Repeating "pseudo-experiments" is expensive CPU-wise (might involve efforts like retraining a neural network, etc).
- What is the smallest number of pseudo-experiments needed to estimate the covariance matrix "reliably"?

Igor Volobouev | How many samples for V estimation? | February 20, 2023 | 2 / 27



The coverage properties of the covariance matrix constructed by sampling techniques depends on the number of samples (as well as some other properties of the matrix)

One needs more samples to estimate more eigenvectors accurately

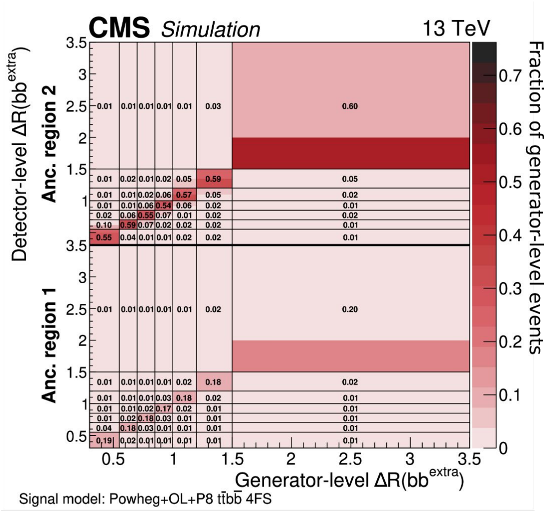
Model Fitting (e.g. likelihood): Profiled Uncertainties

‘Dynamically’ take into account uncertainties + interactions

Nuisance parameters:

part of the model ‘on the same footing’ as parameters of interest

- Correctly propagates non-linearities/interactions



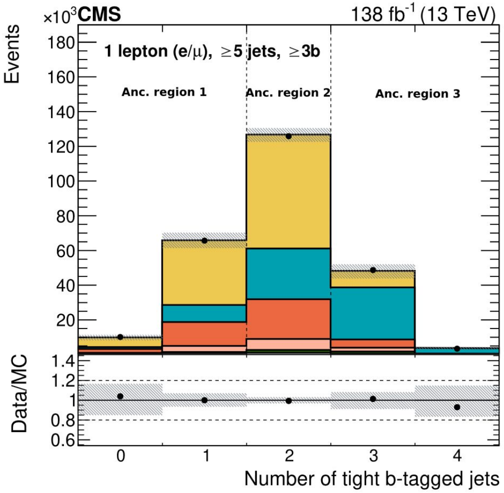
Real advantages start to be seen when

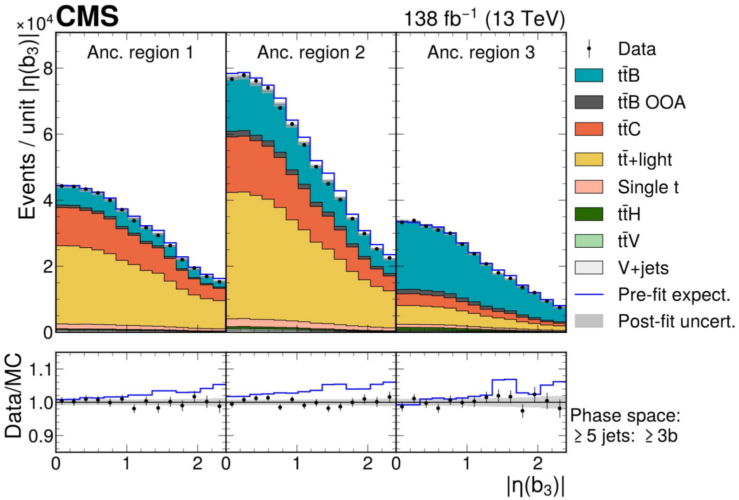
moving beyond ‘square matrix’ unfolding

Profiled Uncertainties

The **advantages of the 'dynamical' nuisance parameter treatment**

When $\mathbf{N_{data} > N_{poi}}$ the subspace orthogonal to that spanned by the POIs acts as a 'control region' which modifies and **constrains other model parameters!**





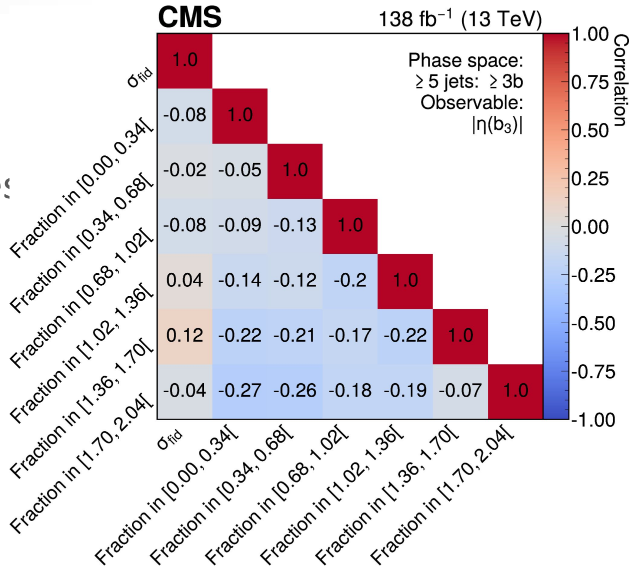
Profiled Nuisance Parameters

Uncertainties typically taken from $-2\Delta\log(\mathscr{L})$ intervals

Covariance typically calculated from inverting second derivatives minimum

✅ Easy to calculate within method
❌ Proper coverage not expected when regularized!

→ Good to check/calculate coverage and bias with toys!



Uncertainty breakdown takes a little bit (really not much!) of extra work

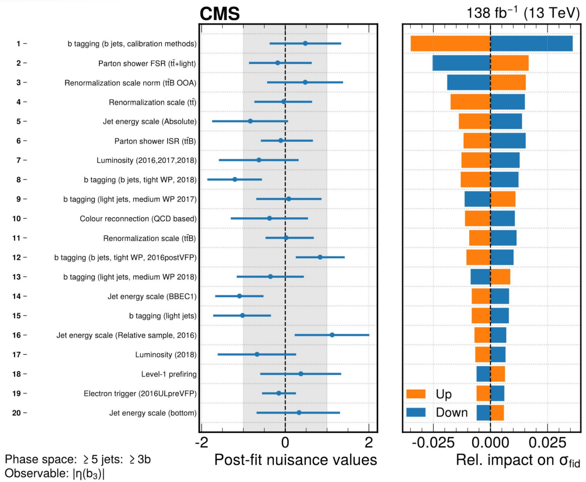
✅ Experiments moving towards publishing full statistical models

→ **Much better than just an uncertainty breakdown**

Profiled Uncertainties: Validation

Investigate the effects of nuisance parameters

Investigate the consistency of the nuisance parameter model



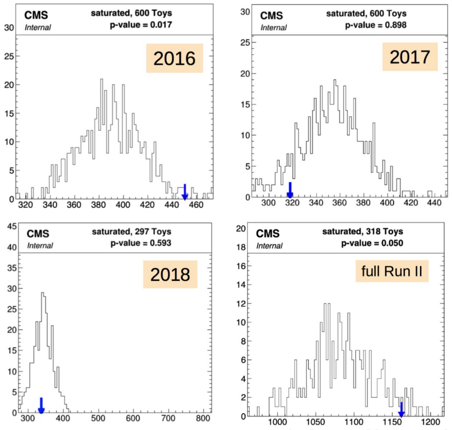


Figure 291: Goodness of fit results using the saturated model for 2016, 2017, 2018 and for the full Run 2 combination.

GOF often not tested in unfolding measurements

When:

$$\#\text{ gen bins} = \#\text{ of detector bins}$$

Freedom in fit equal to freedom in data

Important when doing regularization!

When using more reco information (control regions, eras, channels ….) it provides an additional check of the modelling (and therefore uncertainties)

Profile Likelihood: Some thoughts

The community has some very nicely developed tools for investigating and understand maximum-likelihood fits

Although they are usually not tailored to unfolding (sometimes a little clunky)

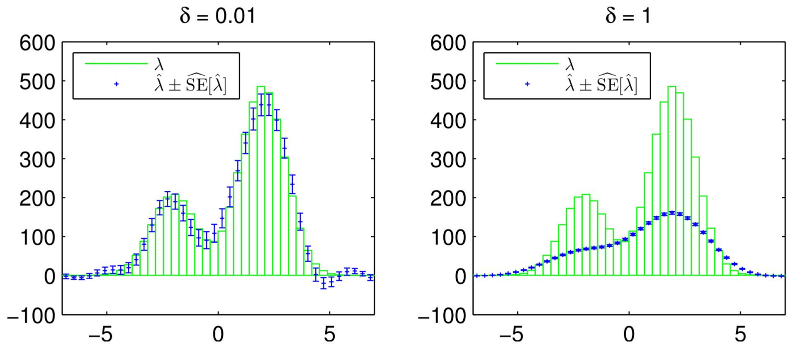
They can provide helpful ways to check uncertainties, validate model, etc…

‘Unfolding Uncertainties’:
Regularization, Bias, Model Dependence

Unfolding Regularization Uncertainties

Taking into account only the 'standard' sources will undercover

→ Need to account for bias



Typically estimate with pseudo-experiments:
Unfold a distribution where the truth is known a priori to estimate the bias

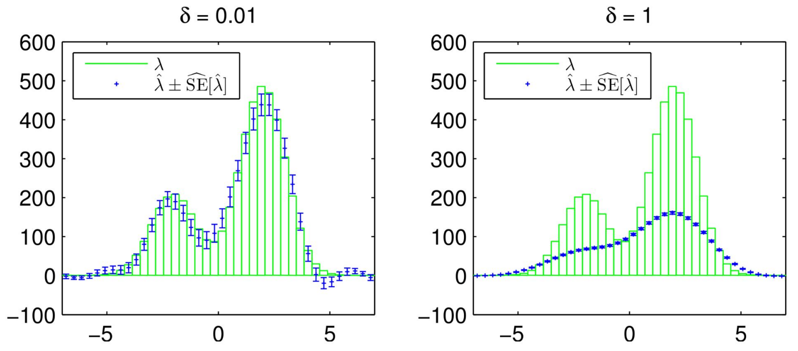
Bias depends on method and **unknown truth distribution**
Have to use good judgement!

Even if the magnitude is a reasonable estimate detailed shape probably isn't!

Unfolding Regularization Uncertainties

Bias depends on the unknown truth

→ Try to estimate bias using samples which have similar expected difference to model as truth



Best practices(?):

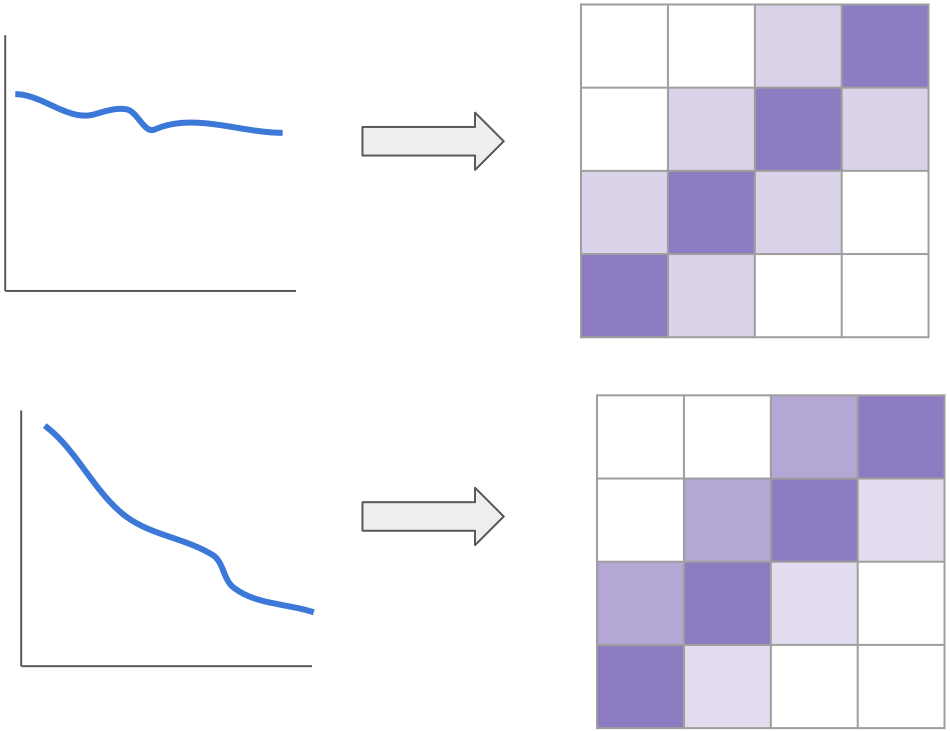
1. Use several different models to calculate bias and include uncertainty
2. Use several more independent models to validate and cross-check

Aim to reduce bias:

Preferably the measurement is dominated by well-understood and well-modelled uncertainties

Model Dependence

Typically the response function depends on the distribution being modeled!

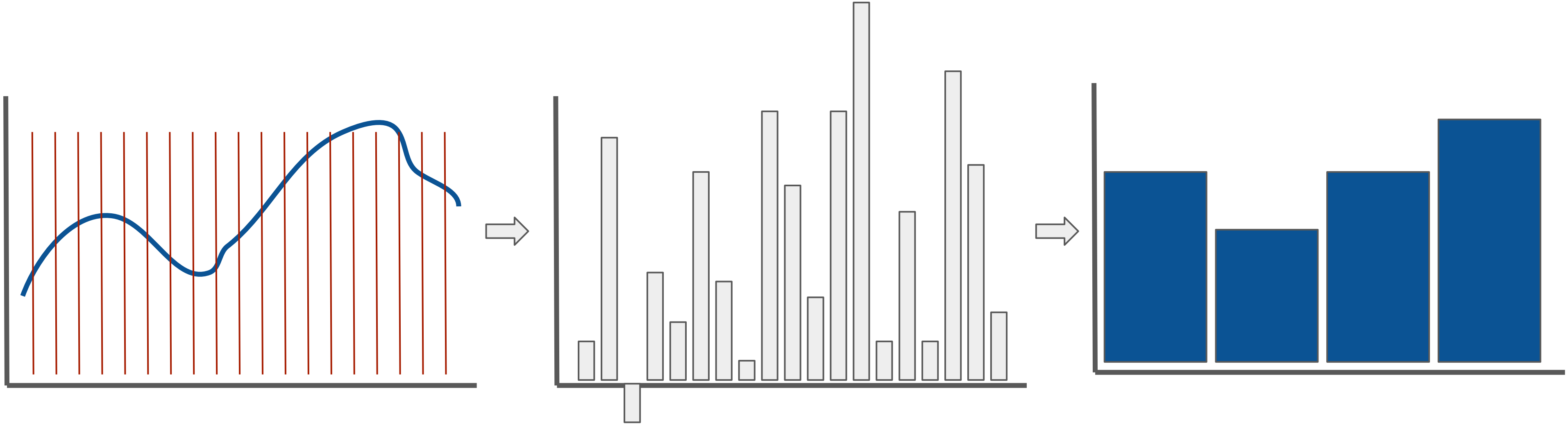


Another form of regularization bias

Don’t just estimate the bias, improve the methods!

Wide-bins-via-fine-bins:

1. Reduce the bias by starting with very fine bins
2. Aggregate fine bins into wide ones to reduce their variances



→ Adding parameters to model to account for shape differences with nominal

Model Dependence Likelihood fits

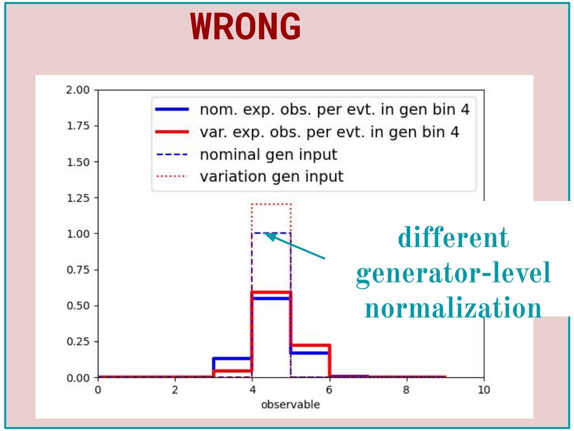
Typically treat via theoretical uncertainties on the signal

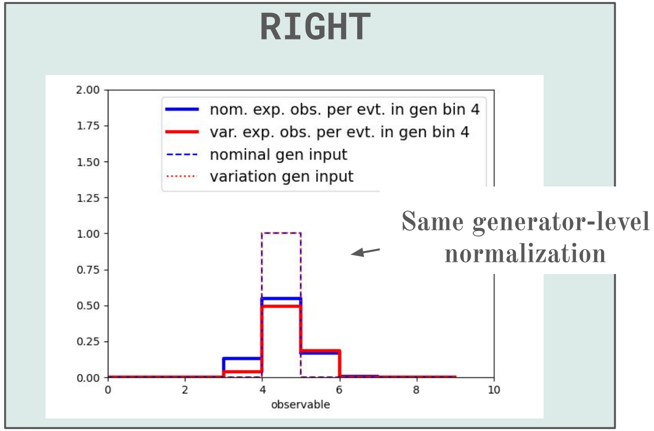
e.g. effects from renormalization and factorization scales: μ_R and μ_F

Normalization changes of each parameter (σ_1, σ_2, ..., σ_n) **should not be included** in the variations.

e.g. Remove differences in $\sigma_1(\mu_R = \mu^{nom})$ and $\sigma_1(\mu_R = \mu^{var})$ when providing migration matrices/efficiencies/templates

Detector-level **changes due to μ_R for a fixed value of σ_i still need to be taken into account**
(can effect acceptance, efficiency, migrations)





Model Dependence in Likelihood fits

As compared to wide-bin-via-fine-bin using theory uncertainties has more model assumptions. Good to check + validate them:

1. Increase the number of bins at detector level
2. Check nuisance parameter pulls, goodness of fit of distributions

Are the theoretical variations able to explain the observed data patterns?

‘New Method’:
Incorporating Uncertainties

New Ideas in unfolding

Many new ideas in unfolding, particularly with Machine Learning

A crucial step in going from ‘nice idea’ to usable result is uncertainty estimation

- Sometimes dedicated new ideas can be used
- Sometimes existing methods can be re-used or re-adapted

Sometimes new ideas can help reduce uncertainties, e.g.:

simplify including more data features

→ more complete modelling of detector response

→ less dependence on assumed distributions

Example: Uncertainties with Omnifold

Recent minimum-bias event-shape measurement from CMS using omnifold:
http://cds.cern.ch/record/2899591?ln=en

Prefer to use toy-based uncertainty estimation (don't assume linearity)
Downside: computationally very expensive

Problem: requires per-event parameterization of nuisances, but most uncertainties come from separate MC samples

Example: Uncertainties with Omnifold

New Simulated sample
→Use it for unfolding

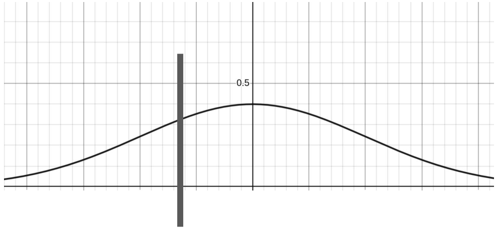
$W_1 = (0.9)\cdot(1.15)$

$W_2 = (0.95)\cdot(1.3)$

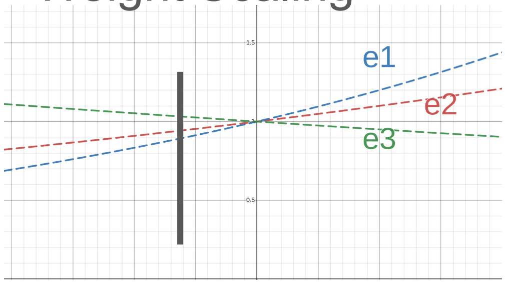
$W_3 = (1.05)\cdot(0.85)$

Syst. 1

Nuisance PDF



Weight Scaling

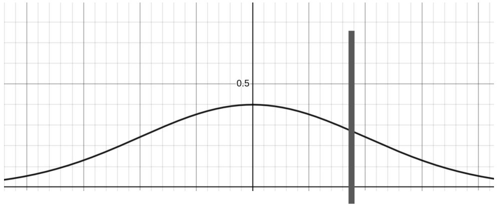


Nuisance value

. . .

Syst. 2

Nuisance PDF



Weight Scaling

Nuisance value

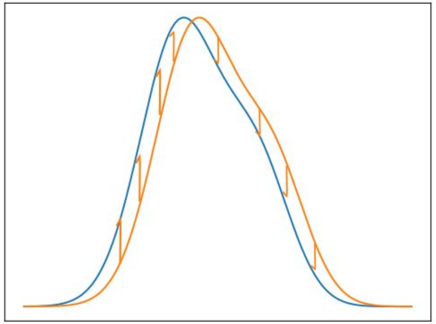
Example: Uncertainties with Omnifold

Q: How to apply weighting function between different MC samples?

A: Use Machine-learning based reweighting

1. Train a classifier to distinguish two samples: **A** and **B**
   a. The classifier should ‘learn the likelihood ratio’

1d-slice for visualization



Reweight

After the reweighting has been derived (And validated!) for the 1σ template – everything follows as in the binned case

Final Thoughts

Final Thoughts

None of our methods are perfect, but we should keep trying to improve them

- Reduce reliance on assumptions
- Improve measurements by including more information
- Carefully check and validate our models: bias + coverage
- Provide better public information (uncertainty breakdowns, likelihoods)
- Continue to develop new methods and tools

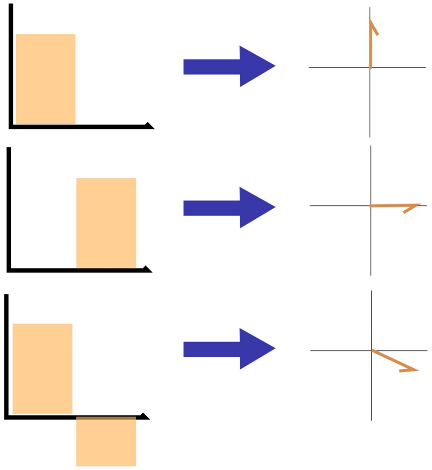
- Larger accurate uncertainties are (much) more useful than smaller inaccurate ones!

Backup

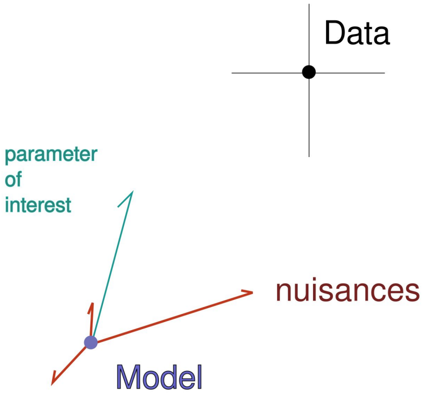
Profiled Uncertainties

When $N_{poi} < N_{data}$ interesting things start to happen

Consider 'linearized' simplification



N-binned histograms are N-dimensional vectors



Nuisance parameters have linear effects on prediction

Components of nuisance vectors parallel to the space spanned by the POIs contribute to the POI uncertainty

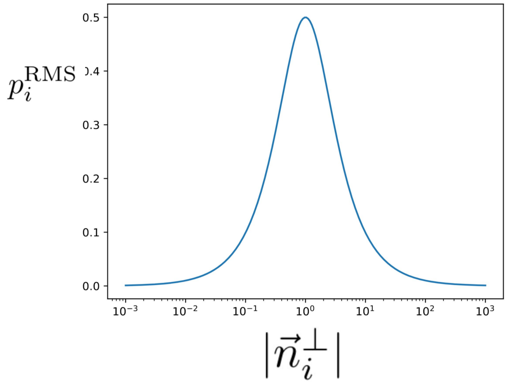
Components of nuisance vectors orthogonal to the space spanned by the POIs are constrained

Something to watch out for!

Orthogonal (to POI) subspace ⟶ pulls + constraints parameters

Parallel (to POI) subspace ⟶impacts POI values

The effect of random noise in the orthogonal subspace can impact the POI estimate via the nuisance parameter, depending on the sizes and its parallel ($\vec{n}^{/\!/}$) and perpendicular ($\vec{n}^{\perp}$) components**.



$$\Delta\mathrm{POI}^{\mathrm{RMS}} = p_i^{\mathrm{RMS}} \vec{n}_i^{||} = \frac{|\vec{n}_i^{||}||\vec{n}_i^{\perp}|}{1 + \vec{n}_i^{\perp 2}}$$

* the vector space metric is defined such that the statistical uncertainties in any bin are equal to 1

and the nuisance parameter vector corresponds to the 1σ effect from that parameter

*This is derived ignoring interplay with other n.p.

What contributes to the pull of a nuisance?

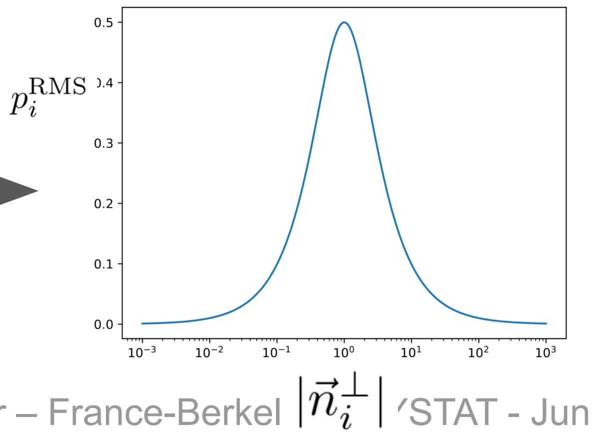
➢ Starting from the (linearized) log-likelihood we have the equation

$$\frac{\partial \mathrm{nll}}{\partial p_i} = 0 \implies p_i = \frac{\vec{n}_i^{\perp} \cdot (\vec{\Delta}^{\perp} - \sum_{j \neq i} p_j \vec{n}_j^{\perp})}{1 + \vec{n}_i^{\perp 2}}$$

The expected pull from statistical fluctuations in the data is largest for nuisance vectors of length 1

Modelling $\vec{\Delta} - \Sigma p_j \vec{n}_j$ as white noise gives

$$p_i^{\mathrm{RMS}} \approx \frac{|\vec{n}_i^{\perp}|}{1 + \vec{n}_i^{\perp 2}}$$



Intuitive explanation:

- Length >> 1 will be constrained by statistics
- Length << 1 are too costly to pull