Science and
Technology
Facilities Council

Particle Physics

Hands-on Statistics

Tim Adye, Will Buttinger

Rutherford Appleton Laboratory

PPD Advanced Graduate Lectures
25th May 2023

ATLAS

Introduction

- This is not a complete statistics lecture
  - Instead, I hope to introduce some of the statistical techniques used in Particle Physics that may not have been covered by more general statistics courses, and give some hints on how to use them.
- We will only discuss techniques specifically used in Particle Physics today, notably:
  1. Frequentist statistics
     - Bayesian used in most other fields
  2. profile likelihood ratio in the form developed for the LHC (and used elsewhere)
     - you are probably already familiar with the other common method, least squares (χ^2) fit
  3. CLs limits on rates, cross-sections, etc.
     - not really used outside our field
  4. Unfolding histograms
     - also used in scientific image processing (AKA "deconvolution" or "unsmearing")
- We will not discuss the following techniques, which are more commonly taught in statistics courses
  1. combination of results (BLUE etc)
     - I will mention Likelihood combination
  2. goodness-of-fit (χ^2, KS test, etc)
  3. ... or any techniques primarily used on event data, before the final statistical interpretation
     - multivariate discrimination, machine learning, sPlots, etc.

Lecture plan

- Building a model
  1. PDF ⊗ data → Likelihood
  2. Asimov dataset

- Testing a model
  - examples from LHC Run1 Higgs measurements → all three stages:
  3. Measurement

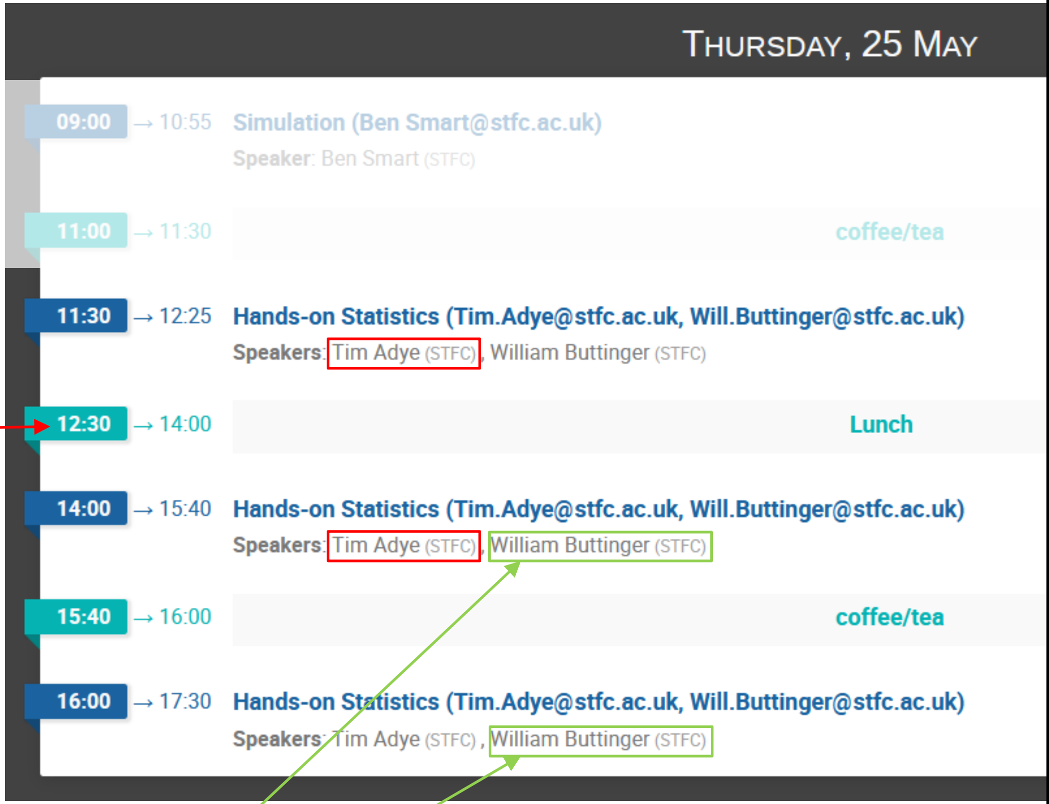
➢ break for lunch here?

  4. Discovery
  5. Exclusion
     - full example in Will’s tutorial

- Presenting results without a model
  6. Unfolding

  7. Summary

- Will be followed by a Hands-on Tutorial this afternoon, run by Will



Thursday, 25 May

09:00 → 10:55 Simulation (Ben Smart@stfc.ac.uk)
Speaker: Ben Smart (STFC)

11:00 → 11:30 coffee/tea

11:30 → 12:25 Hands-on Statistics (Tim.Adye@stfc.ac.uk, Will.Buttinger@stfc.ac.uk)
Speakers: Tim Adye (STFC), William Buttinger (STFC)

12:30 → 14:00 Lunch

14:00 → 15:40 Hands-on Statistics (Tim.Adye@stfc.ac.uk, Will.Buttinger@stfc.ac.uk)
Speakers: Tim Adye (STFC), William Buttinger (STFC)

15:40 → 16:00 coffee/tea

16:00 → 17:30 Hands-on Statistics (Tim.Adye@stfc.ac.uk, Will.Buttinger@stfc.ac.uk)
Speakers: Tim Adye (STFC), William Buttinger (STFC)

Model building

PDF, dataset, and likelihood

- All the statistical tests we will be considering are based on the likelihood

$$L(\boldsymbol{\mu},\boldsymbol{\theta}) = \prod_c \prod_i P_c(x_i|\boldsymbol{\mu},\boldsymbol{\theta}) \cdot \prod_j C_j(g_j|\theta_j)$$

1. $L(\boldsymbol{\mu},\boldsymbol{\theta})$ is a function of one or more parameters of interest ($\boldsymbol{\mu}$), as well as other nuisance parameters ($\boldsymbol{\theta}$)
2. $P_c(x_i|\boldsymbol{\mu},\boldsymbol{\theta})$ is the probability density function (PDF) for channel c, evaluated for each member of the dataset, x_i
   - The use (or not) of the parameters, $\boldsymbol{\mu},\boldsymbol{\theta}$, in the different channels determines how they are constrained by data
   - eg. for binned data in histogram h, with bins, i, $P_h(n_i|\boldsymbol{\mu},\boldsymbol{\theta}) = \mathrm{Poisson}(n_i|\nu_i(\boldsymbol{\mu},\boldsymbol{\theta}))$
3. $C_j(g_j|\theta_j)$ are additional PDFs that do not depend on the data
   - eg. constraint terms for systematic uncertainties, $C_j\left(g_j\middle|\theta_j\right) = \mathrm{Gaussian}(g_j|\theta_j,\sigma_j)$

- Bear in mind:
  - PDFs ($P_c(x)$ and $C_j(g)$) must be normalised to 1, or a constant independent of $\boldsymbol{\mu},\boldsymbol{\theta}$
    - The likelihood, on the other hand, is not normalised
  - The absolute value of the likelihood ($L(\boldsymbol{\mu},\boldsymbol{\theta})$) is irrelevant, only changes WRT $\boldsymbol{\mu},\boldsymbol{\theta}$
  - It is usually used as $-\ln L$, or more commonly, $-2\ln L$

$$-2\ln L(\boldsymbol{\mu},\boldsymbol{\theta}) = \sum_c \sum_i -2\ln P_c(x_i|\boldsymbol{\mu},\boldsymbol{\theta}) + \sum_j -2\ln C_j(g_j|\theta_j)$$

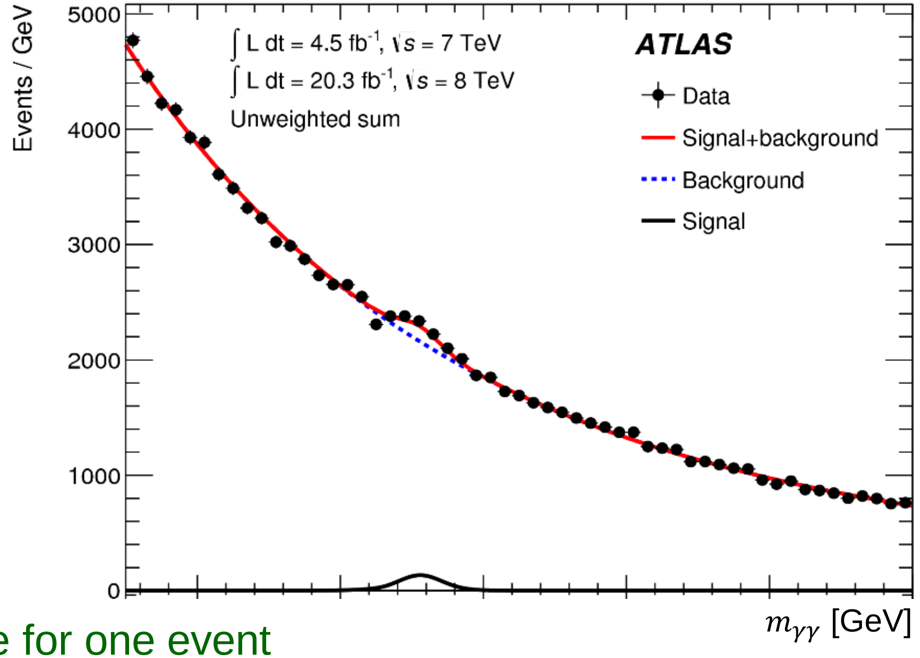
    - maximum likelihood is at minimum of $-2\ln L$
    - in the Asymptotic limit, $-2\ln L$ is distributed like a χ^2

Terminology

PRD 90 (2014) 112015

$$L(\boldsymbol{\mu},\boldsymbol{\theta}) = \prod_c \prod_i P_c(x_i|\boldsymbol{\mu},\boldsymbol{\theta}) \cdot \prod_j C_j(g_j|\theta_j)$$

- Model PDF, function of
  - observables, x_i, or $m_{\gamma\gamma}$
  - parameters of interest (POIs), $\boldsymbol{\mu}$, eg. $\mu_{\gamma\gamma}$ and/or m_H
  - nuisance parameters (NPs), $\boldsymbol{\theta}$
    - Mostly give systematic uncertainties
      - eg. luminosity, efficiencies, energy scale, theory uncertainties (signal and background)



- Dataset
  - Set of entries, each containing values of some of the observables, x_i
    - binned dataset: each entry contains the contents of a bin
    - unbinned dataset: each entry contains the measured value of the observable for one event
    - Datasets often combined for different channels, even different observables, or combined binned and unbinned
  - Also associated global observables [1] that are common to all entries
    - Many of these give central value of a systematic uncertainty used in the constraint term
- A likelihood fit, usually to $-2\ln L$
  - minimises the likelihood with respect to floating parameters
  - depending on the statistical test, some POI/NPs may be fixed

[1] Global observables are not currently part of the RooFit dataset, but should be logically associated

RooFit

- RooFit is a tool for creating models
  - RooAbsPdf: base class for PDFs. Will often be constructed from many PDF types.
    - eg. RooGaussian, RooProdPdf, RooSimultaneous
    - these are functions of each other, and of RooRealVar parameters that can be mapped to fit parameters
    - can be constructed directly from C++ or Python, or via a “factory” from a specification
      - eg. SUM::model (f*RooGaussian::g(x,m[0],1), RooChebychev::c(x,{a0[0.1],a1[0.2],a2[-0.3]}))
  - RooAbsData: abstract dataset type. Can hold binned and/or unbinned data
  - RooStats::ModelConfig (optional): holds configuration information for a single model
    - PDF, POIs, NPs, observables, etc
  - RooWorkspace: container for PDFs, datasets, and ModelConfigs
    - This can be saved to a workspace.root file to allow separate statistical analysis
    - everything needed should be stored here, allowing sharing, combining, archiving

- RooFit also provides fitting and basic statistical analysis tools
  - RooNLLVar: $-\ln L$ constructed from PDF and dataset
  - RooMinimizer: uses Minuit to minimise RooNLLVar for specified parameters

Higher-level tools

- **2007**: RooFit package in ROOT allows building models and fitting to data using the venerable MINUIT [CERN, 1975–] minimization algorithm
- **2008**: RooStats added to ROOT to provide limit-setting functionality
  - a range of different techniques (Frequentist, Hybrid, Asymptotic, and some basic support for a fully Bayesian approach)
  - also introduced the HistFactory model specification
    - a way of defining a binned model based on input ROOT histograms and XML metadata files
- **2009–2018**: HistFactory and RooStats heavily used via a multitude of higher-level toolkits
  - Fitting large models took longer and longer [1].
    - Could this be improved by replacing MINUIT with other minimization algorithms running on GPUs?
- **2018**: pyhf and zfit released. They can both exploit GPUs.
  - pyhf: pure-Python implementation of the HistFactory specification
  - zfit: unbinned models minimized using TensorFlow
- **2019–2021**: exploitation and comparison of the new techniques revealed that:
  - GPUs can help unbinned models (zfit), but not so much binned models (pyhf)
  - pyhf's JSON format for HistFactory was a genuine improvement over the previous XML+ROOT format
  - improving PyROOT makes using RooFit just as easy from Python or C++
- **2022**: JSON-format HistFactory is added to RooFit
  - GPU support also in development for RooFit
- **2023**: xRooFit – a new experimental high-level API for RooFit
  - "xRooFit is to RooFit as Keras is to TensorFlow“

[1] At the time of the Higgs discovery, Higgs model fitting with RooStats was the largest single analysis user of ATLAS Grid CPU.

Advice from Will: unless your research is specifically on optimizing fitting for statistical analysis, you are best-placed if you *stay as close to RooFit as possible*, safe in knowledge that any worthwhile improvements developed elsewhere will make their way into RooFit in time

Asimov dataset

- An Asimov dataset [1] is generated for a particular set of model parameters such that the maximum likelihood best-fit value of all those parameters are equal to their generated values.
  - ie. maximising $L_A(\mu,\boldsymbol{\theta}|\mu_0,\boldsymbol{\theta}_0)$ will yield $\hat{\mu}=\mu_0,\widehat{\boldsymbol{\theta}}=\boldsymbol{\theta}_0$
  - When used in a statistical test, it will return the result expected from that model configuration
    - eg. p_0 calculated using Asimov dataset generated with $\mu=0$ will return the p-value expected from no signal

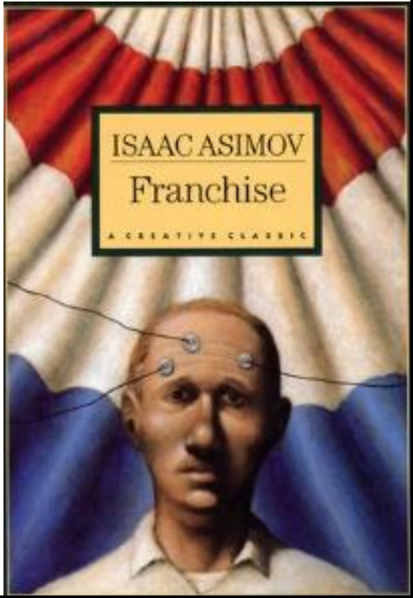
- Asimov datasets are built as binned datasets, in which the event count in each bin is set to the expected event yield for the chosen model parameters.
  - For unbinned models, a binned distribution is generated with chosen binning fine enough to reproduce all significant features of the model.
  - Note this means the Asimov dataset can look different from data or toy datasets: fractional bin contents or unbinned→binned

- For RooFit models:

```
dataset = RooStats::AsymptoticCalculator::GenerateAsimovData (pdf, observables);
```

[1] Named for SF author, Isaac Asimov, whose 1955 short story, *Franchise*, envisaged the 2008 US Presidential Election decided by one voter representative of the entire electorate. [arXiv:1007.1727]

As an Asimov fan of old, this name makes me very happy.

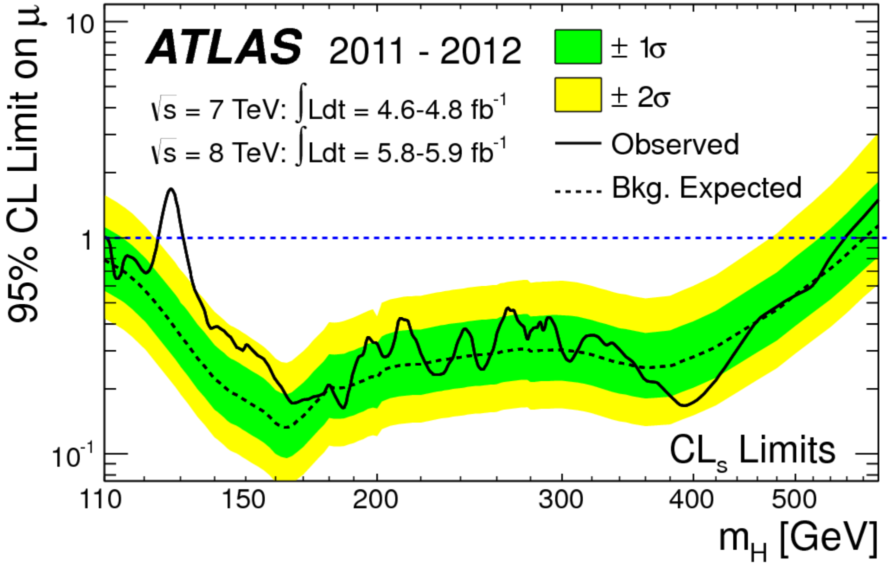


Statistical tests

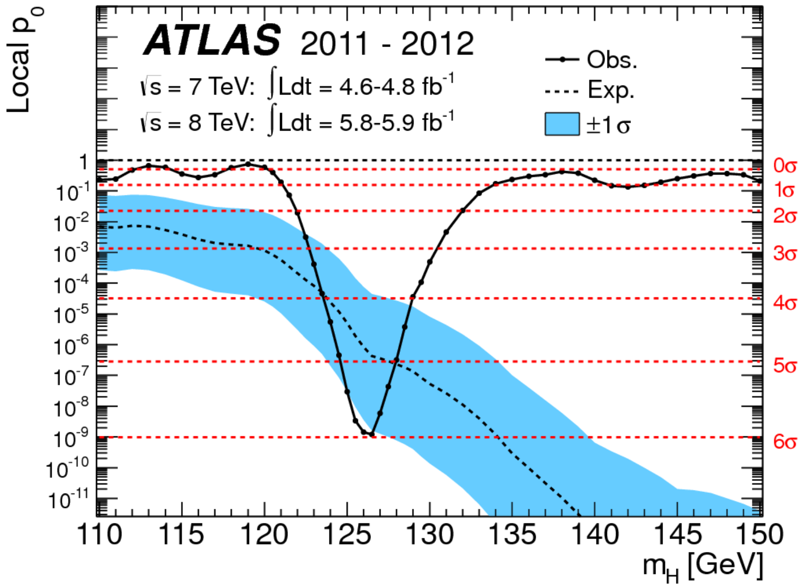
The three ages of discovery

PLB 716 (2012) 1

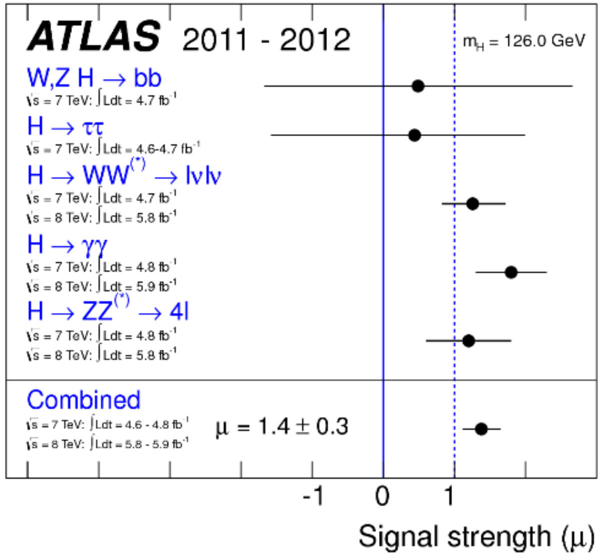
1. **Exclusion**: CLs

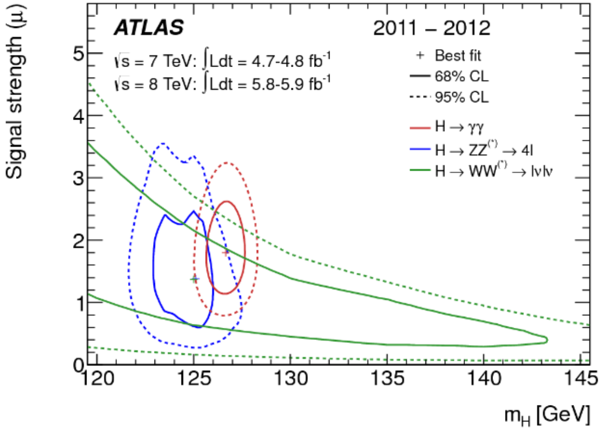


2. **Discovery**: p_0



3. **Measurement**: $\hat{\mu} \pm \sigma$
or more generally,
confidence intervals





Measurement

Measurement: profile likelihood ratio

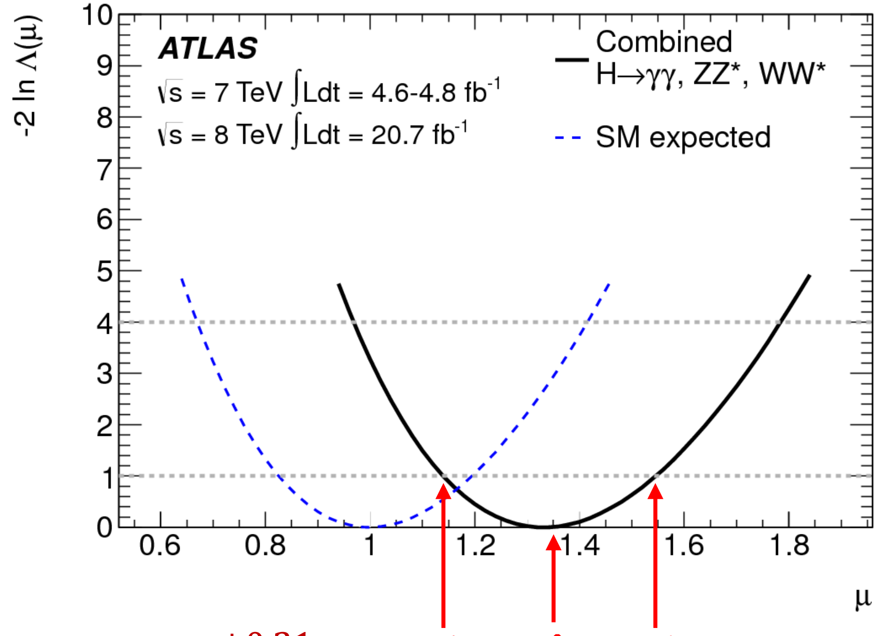
PLB 726 (2013) 88

- The likelihood is a function of our parameters of interest (POI), here a single μ, and various nuisance parameters (NP), $\boldsymbol{\theta}$: $L(\mu, \boldsymbol{\theta})$.
  - Note that the $\boldsymbol{\theta}$ are often dependent on μ.
- We form the profile likelihood ratio as: $\Lambda(\mu) = \dfrac{L(\mu, \hat{\hat{\boldsymbol{\theta}}}(\mu))}{L(\hat{\mu}, \hat{\boldsymbol{\theta}})}$

maximise $L\big(\mu, \boldsymbol{\theta}(\mu)\big)$ for all $\boldsymbol{\theta}(\mu)$ with specified μ

maximise $L(\mu, \boldsymbol{\theta})$ for all $\mu, \boldsymbol{\theta}$ (MLE)

  - $\Lambda(\mu)$ can be evaluated with two fits:
    1. $\hat{\mu}$ and $\hat{\boldsymbol{\theta}}$ are the “best fit” (maximum likelihood estimate, MLE) values of μ and $\boldsymbol{\theta}$
    2. $\hat{\hat{\boldsymbol{\theta}}}(\mu)$ are the “conditional best fit” values for all the NPs at a given, specified, μ.
- Plot $-2\ln\Lambda(\mu)$ against μ
  - Minimum is at $-2\ln\Lambda(\hat{\mu}) = 0$ (by definition)
  - In the asymptotic limit (large N),
    - this will be distributed like a χ^2_1 distribution
      - or χ^2_n for n POIs
    - so 68% confidence interval is the range where $-2\ln\Lambda(\mu) < 1$



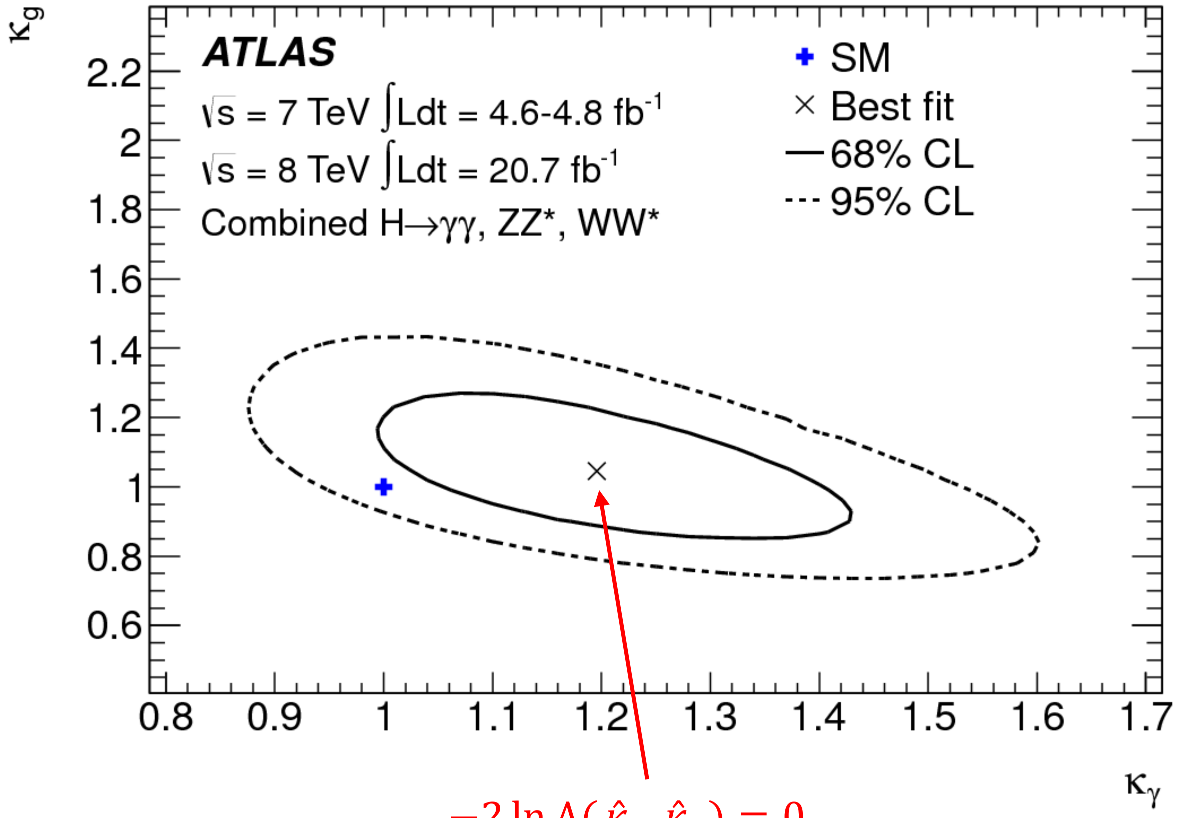
$\mu = 1.33^{+0.21}_{-0.18}$ -1σ $\hat{\mu}$ $+1\sigma$

Measurement: 2D contours

PLB 726 (2013) 88

[1] $D^{-1}(\chi_n^2(p))$ = ROOT::Math::chisquared_quantile (p, n)

- For multiple POIs
  - calculate $-2\ln\Lambda(\boldsymbol{\mu})$ for all points on a grid and
  - draw contours for regions $-2\ln\Lambda(\boldsymbol{\mu}) < D^{-1}(\chi_n^2)$,
    - where $D^{-1}(\chi_n^2)$ is the inverse of the cumulative χ_n^2 distribution, for n POIs. [1]
    - 2D contours:
      - $D^{-1}(\chi_2^2(68\%))$ = 2.30
      - $D^{-1}(\chi_2^2(95\%))$ = 6.18

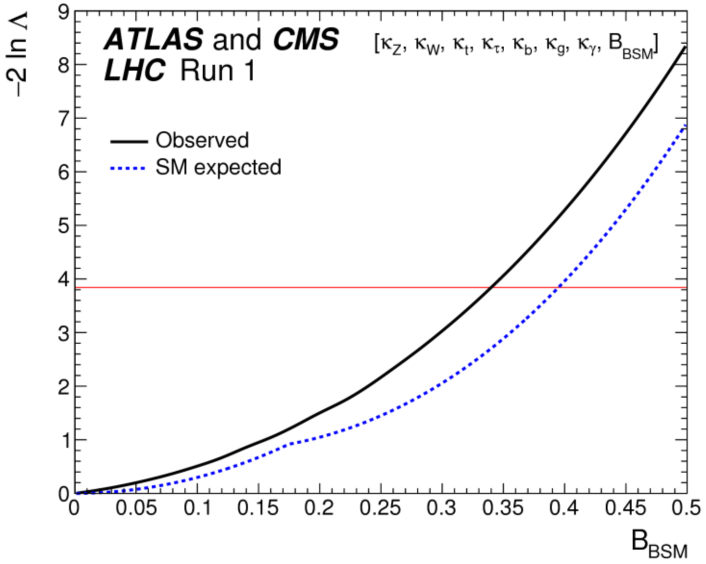


$-2\ln\Lambda(\hat{\kappa}_\gamma, \hat{\kappa}_g) = 0$

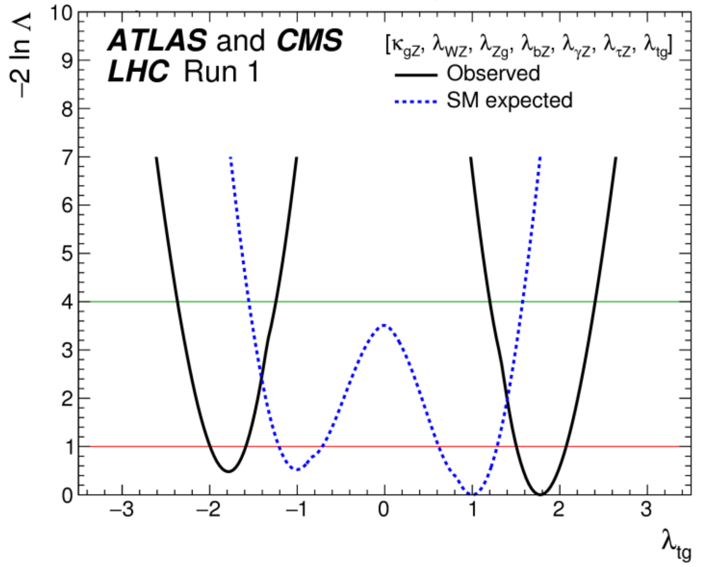
Measurement: variations on a theme

JHEP 08 (2016) 045

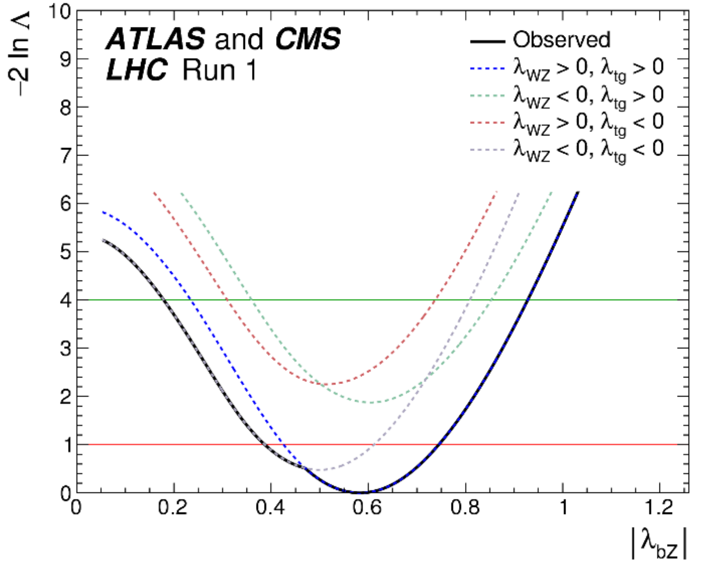
95% confidence interval with ($B_{BSM}\geq0$) bound: $B_{BSM} < 0.34$



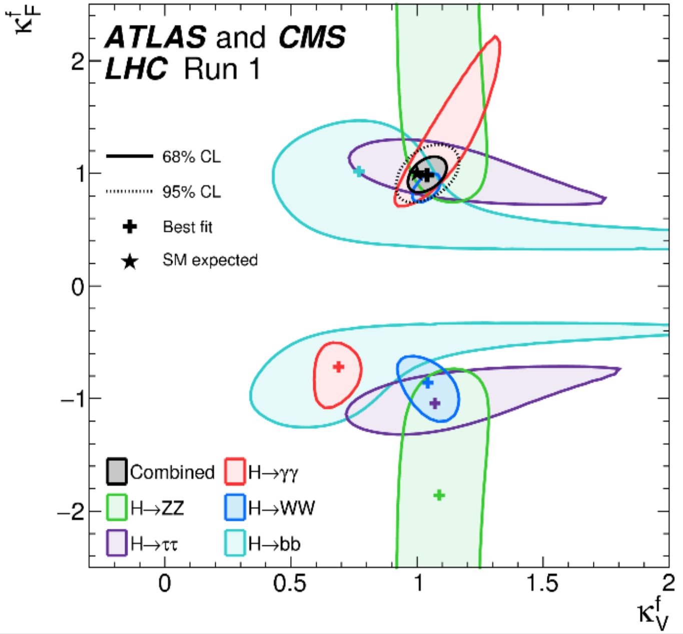
disjoint confidence interval: λ_{tg} = [−2.00, −1.59] ∪ [1.50, 2.07]



kink due to different sign combinations of profiled NPs: $|\lambda_{bZ}| = 0.58^{+0.16}_{-0.20}$

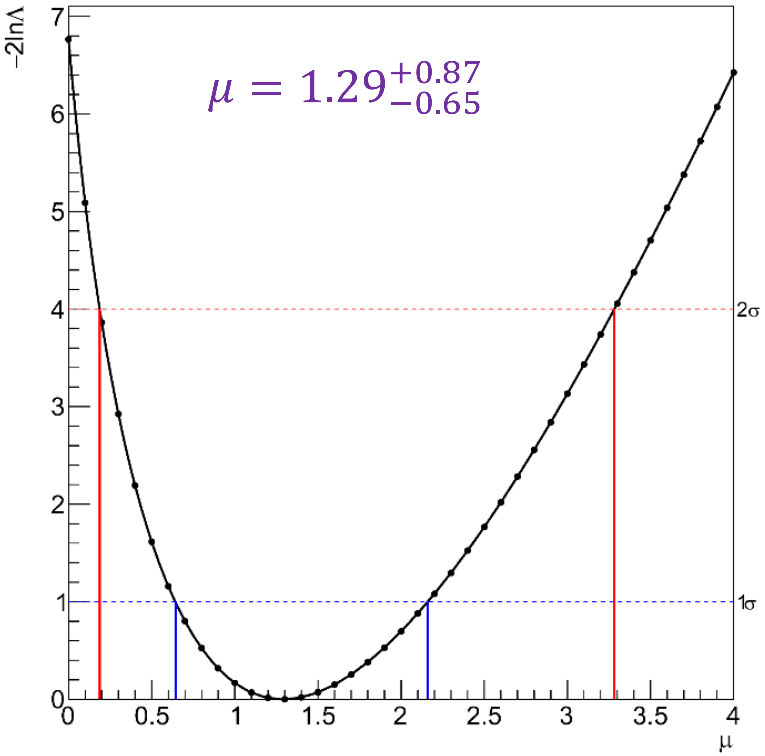


multiple contours for different channels and their combination



Measurement: scanning the likelihood curve

- To calculate a single PLR, require two fits:
  - $-2\ln\Lambda(\mu) = -2\ln\frac{L\left(\mu,\hat{\hat{\boldsymbol{\theta}}}(\mu)\right)}{L(\hat{\mu},\hat{\boldsymbol{\theta}})}$
    $= -2\ln L\left(\mu,\hat{\hat{\boldsymbol{\theta}}}(\mu)\right) + 2\ln L\left(\hat{\mu},\hat{\boldsymbol{\theta}}\right)$
  - The second term is independent of μ, so only needs to be evaluated once
  - … or not at all, if the minimum can be determined from the curve
    - removes ambiguity from the offset calculated in two ways (unconditional vs conditional fits)
    - should be ~quadratic near minimum, so can use a quadratic interpolation of lowest 3 points



- Can (approximately) cross-check the result with the unconditional fit for $-2\ln L\left(\hat{\mu},\hat{\boldsymbol{\theta}}\right)$:
  - $\hat{\mu}$ should agree within the precision of the fit and of the interpolation
  - inverse Hessian at the minimum is the local covariance matrix, so $\sigma_0^2 = H^{-1}(\mu,\mu)$
    - Minuit will calculate (symmetric) errors from the Hessian
      - run with strategy=2, or call Hesse() explicitly.
    - Minuit's Minos() is similar to the curve scan, but without user control or diagnostic plot
  - Example comparison: $\mu = 1.29^{+0.87}_{-0.65}$ (curve) with $\mu = 1.29 \pm 0.73$ (Hessian)

Measurement: technical issues – CPU

- Sometimes significant CPU requirements
  - Time = (likelihood evaluation time) * (number of evaluations to fit) * (number of fits)
  - Mitigations:
    - Simplify likelihood (faster likelihood evaluation)
    - Reduce or combine number of NPs (simplifies likelihood and fewer fit cycles)
    - Use fewer points in scan and interpolate (quadratic or spline)
      - 2D interpolation is more cumbersome
        - ROOT's TGraph2D can do linear interpolation of contours (use GetContourList() to extract)
    - Run different points in parallel, eg. in batch or on the Grid.

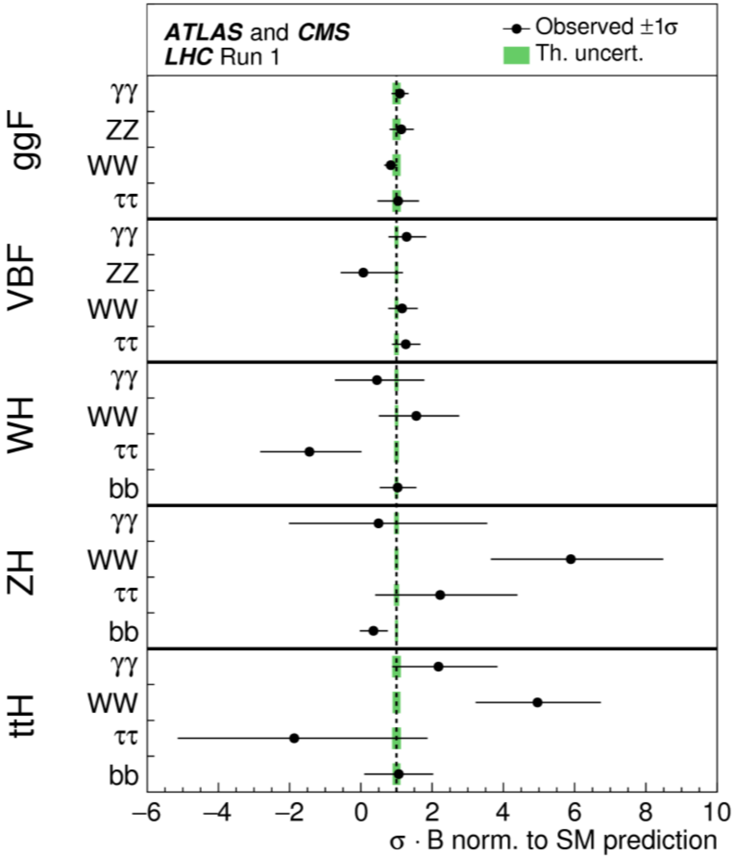
Measurement: technical issues – fitting

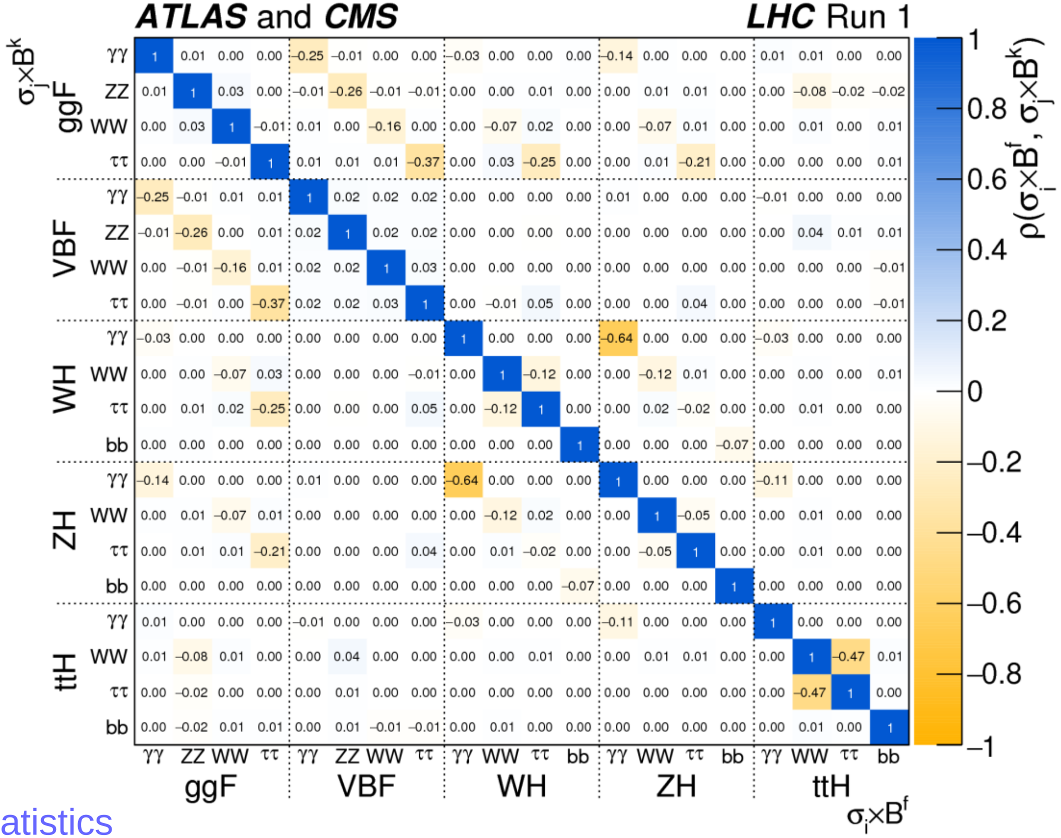
- Fit problems
  1. Fit failures reported by MINUIT (or other minimiser)
     - often due to flat or otherwise non-parabolic minimum
  2. Bumpy curve, kinks, or bad points – even if MINUIT says the fit succeeded
  - Possible causes:
     - Numerical precision in likelihood evaluation
     - Undefined component in likelihood evaluation
        - eg. –ve $\log$ for some observables, in a region of parameter space that the fit strays into
     - Minuit tolerance settings
     - NPs hitting their parameter boundary
        - error estimate will not be correct, even inconsistent
        - parameter errors vs. $\sqrt{V_{ii}}$
     - Some POIs or NPs don’t budge from initial position
     - Minuit can’t “tunnel” from secondary minimum

Measurement: correlation matrix

JHEP 08 (2016) 045

- For ≥ 3 POIs, it is not often practical to show contours
  - requires scanning a large number of points
  - results not easy to visualise
- Another option is to provide the correlation matrix at the best-fit point for all POIs
  - calculate using inverse Hessian $\rho(\mu_1,\mu_2) = H^{-1}(\mu_1,\mu_2)\ /\ \left(H^{-1}(\mu_1,\mu_1)H^{-1}(\mu_2,\mu_2)\right)^{1/2}$
  - but beware that the correlations at the best-fit can be quite different elsewhere





		ggF				VBF				WH				ZH				ttH			
		γγ	ZZ	WW	ττ	γγ	ZZ	WW	ττ	γγ	WW	ττ	bb	γγ	WW	ττ	bb	γγ	WW	ττ	bb
ggF	γγ	1	0.01	0.00	0.00	–0.25	–0.01	0.00	0.00	–0.03	0.00	0.00	0.00	–0.14	0.00	0.00	0.00	0.01	0.01	0.00	0.00
	ZZ	0.01	1	0.03	0.00	–0.01	–0.26	–0.01	–0.01	0.00	0.00	0.01	0.00	0.00	0.01	0.01	0.00	0.00	–0.08	–0.02	–0.02
	WW	0.00	0.03	1	–0.01	0.01	0.00	–0.16	0.00	0.00	–0.07	0.02	0.00	0.00	–0.07	0.01	0.00	0.00	0.01	0.00	0.01
	ττ	0.00	0.00	–0.01	1	0.01	0.01	0.01	–0.37	0.00	0.03	–0.25	0.00	0.00	0.01	–0.21	0.00	0.00	0.00	0.00	0.01
VBF	γγ	–0.25	–0.01	0.01	0.01	1	0.02	0.02	0.02	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	–0.01	0.00	0.00	0.00
	ZZ	–0.01	–0.26	0.00	0.01	0.02	1	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.01	0.01
	WW	0.00	–0.01	–0.16	0.01	0.02	0.02	1	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	–0.01
	ττ	0.00	–0.01	0.00	–0.37	0.02	0.02	0.03	1	0.00	–0.01	0.05	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	–0.01
WH	γγ	–0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1	0.00	0.00	0.00	–0.64	0.00	0.00	0.00	–0.03	0.00	0.00	0.00
	WW	0.00	0.00	–0.07	0.03	0.00	0.00	0.00	–0.01	0.00	1	–0.12	0.00	0.00	–0.12	0.01	0.00	0.00	0.00	0.00	0.01
	ττ	0.00	0.01	0.02	–0.25	0.00	0.00	0.00	0.05	0.00	–0.12	1	0.00	0.00	0.02	–0.02	0.00	0.00	0.01	0.00	0.00
	bb	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1	0.00	0.00	0.00	–0.07	0.00	0.00	0.00	0.00
ZH	γγ	–0.14	0.00	0.00	0.00	0.01	0.00	0.00	0.00	–0.64	0.00	0.00	0.00	1	0.00	0.00	0.00	–0.11	0.00	0.00	0.00
	WW	0.00	0.01	–0.07	0.01	0.00	0.00	0.00	0.00	0.00	–0.12	0.02	0.00	0.00	1	–0.05	0.00	0.00	0.01	0.00	0.00
	ττ	0.00	0.01	0.01	–0.21	0.00	0.00	0.00	0.04	0.00	0.01	–0.02	0.00	0.00	–0.05	1	0.00	0.00	0.01	0.00	0.00
	bb	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	–0.07	0.00	0.00	0.00	1	0.00	0.00	0.00	0.00
ttH	γγ	0.01	0.00	0.00	0.00	–0.01	0.00	0.00	0.00	–0.03	0.00	0.00	0.00	–0.11	0.00	0.00	0.00	1	0.00	0.00	0.00
	WW	0.01	–0.08	0.01	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.01	0.00	0.00	1	–0.47	0.01
	ττ	0.00	–0.02	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	–0.47	1	0.00
	bb	0.00	–0.02	0.01	0.01	0.00	0.01	–0.01	–0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	1

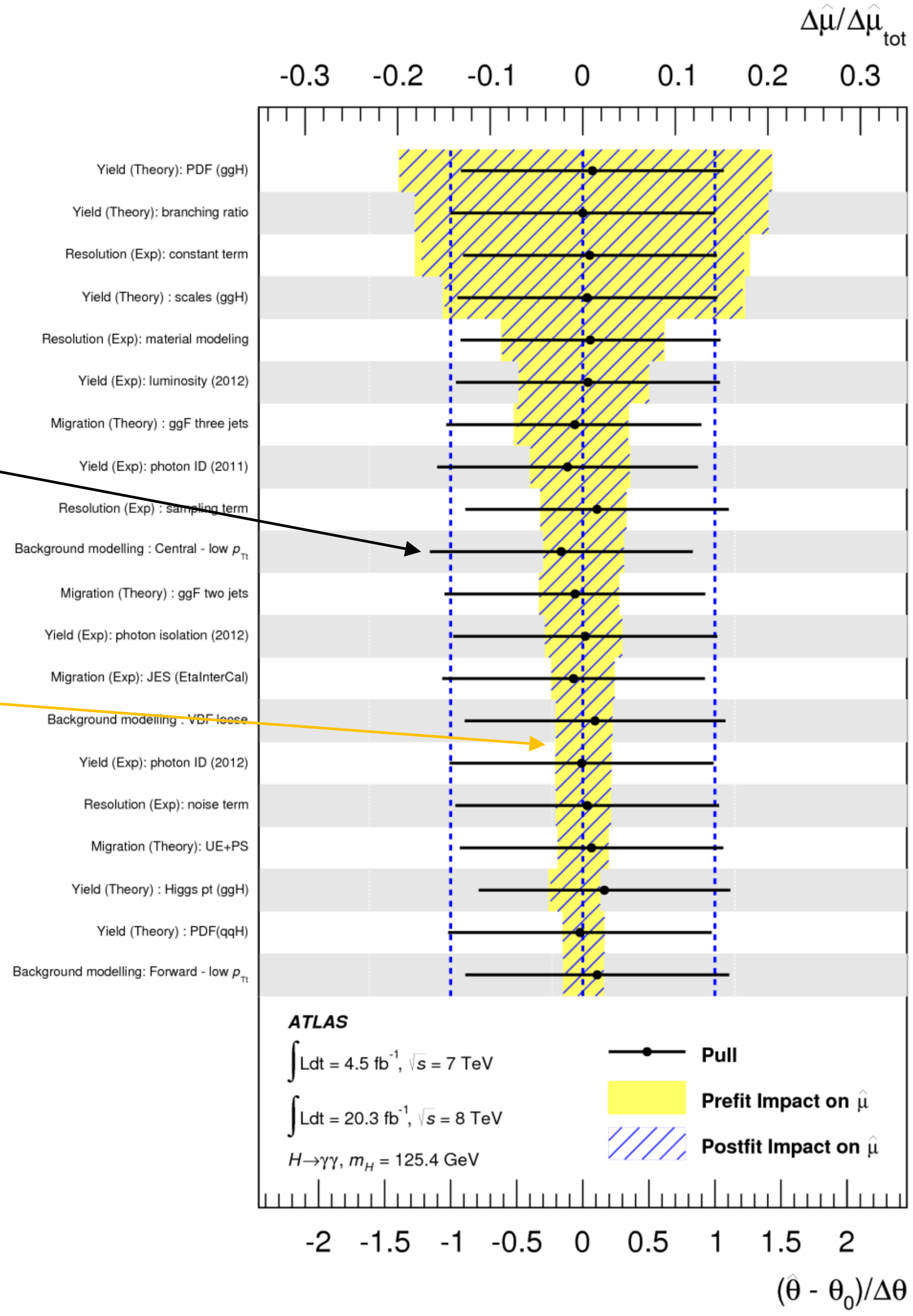
Validation of model

PRD 90 (2014) 112015

- The NPs’ effect on a model can be tested by determining their post-fit pulls and impact on the POI
  - Often (perhaps confusingly) displayed together:

1. NP best-fit value and error
   - relative to nominal, $(\hat{\theta} - \theta_0)/\Delta\theta$, here indicated by blue dotted lines at 0 ± 1.
   - refers to scale at the bottom

2. Impact of NP’s error on POI
   - $\pm\Delta\hat{\mu} = \hat{\hat{\mu}}(\hat{\theta} \pm \sigma_\theta) - \hat{\mu}$
     - important to check relative sign of impact if correlating NPs in a combined workspace
   - can use pre-fit (nominal) and/or post-fit NP errors
   - refers to scale at the top, here relative to the total error, $\Delta\hat{\mu}/\Delta\hat{\mu}_{\text{tot}}$
   - Size of impact indicates importance of each NP



maybe a good place
to break for lunch?

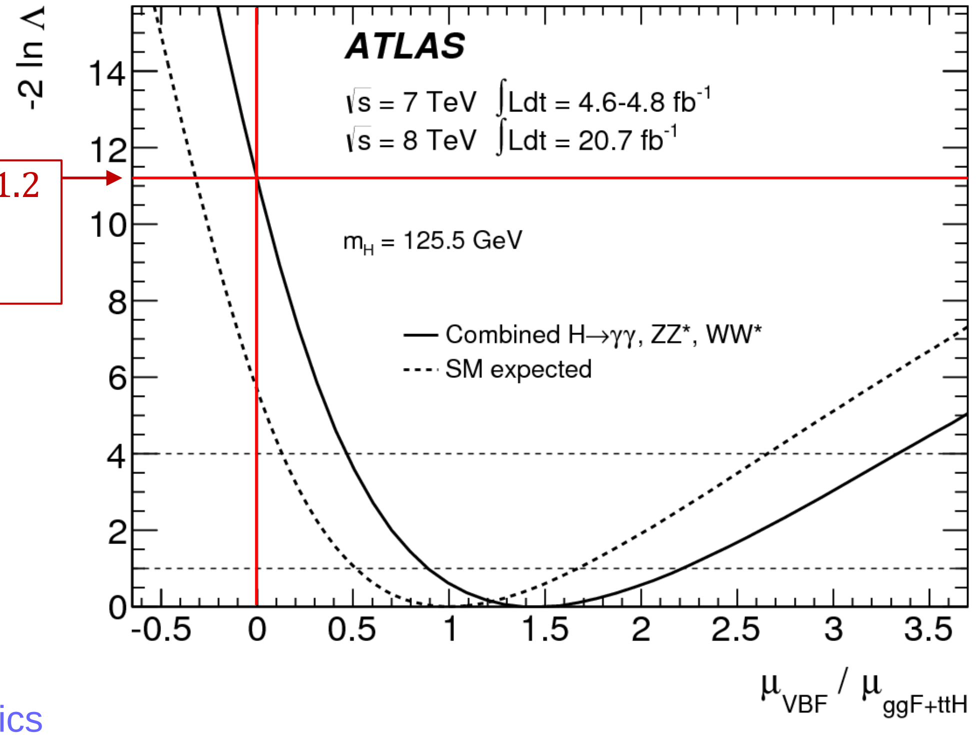
Discovery

Discovery significance

PLB 726 (2013) 88

- In the asymptotic limit (large N), the PLR, $\Lambda(\mu) = \frac{L(\mu,\hat{\hat{\boldsymbol{\theta}}}(\mu))}{L(\hat{\mu},\hat{\boldsymbol{\theta}})}$, gives the compatibility between μ and $\hat{\mu}$ hypotheses.
  - Where μ is a ratio relative to the SM (eg. $\mu = \sigma/\sigma_{\text{SM}}$), we can test
    1. Compatibility with background-only hypothesis: $Z_0 \ = \sqrt{-2\ln\Lambda(\mu=0)}$
    2. Compatibility with SM (1 POI): $Z_{\text{SM}} = \sqrt{-2\ln\Lambda(\mu=1)}$
    3. Compatibility with SM (n POIs): $Z_{\text{SM}} = D^{-1}(\chi_n^2(-2\ln\Lambda(\boldsymbol{\mu})))$
  - Z_μ is the significance ($N\sigma$), which (assuming χ_1^2 for 1 POI) has equivalent p-value, $p_\mu = s\,\Phi(-Z_\mu)$, where
    - $s = 1$ for single-sided test like p_0 [1]
    - $s = 2$ for double-sided test like p_{SM}
    - $\Phi(Z)$ is the Gaussian CDF [2]

- p_0 interpreted as the significance of a signal, relative to a background-only hypothesis



$-2\ln\Lambda = 11.2$
$Z_0 \ = 3.3\sigma$
$p_0 \ = 0.04\%$

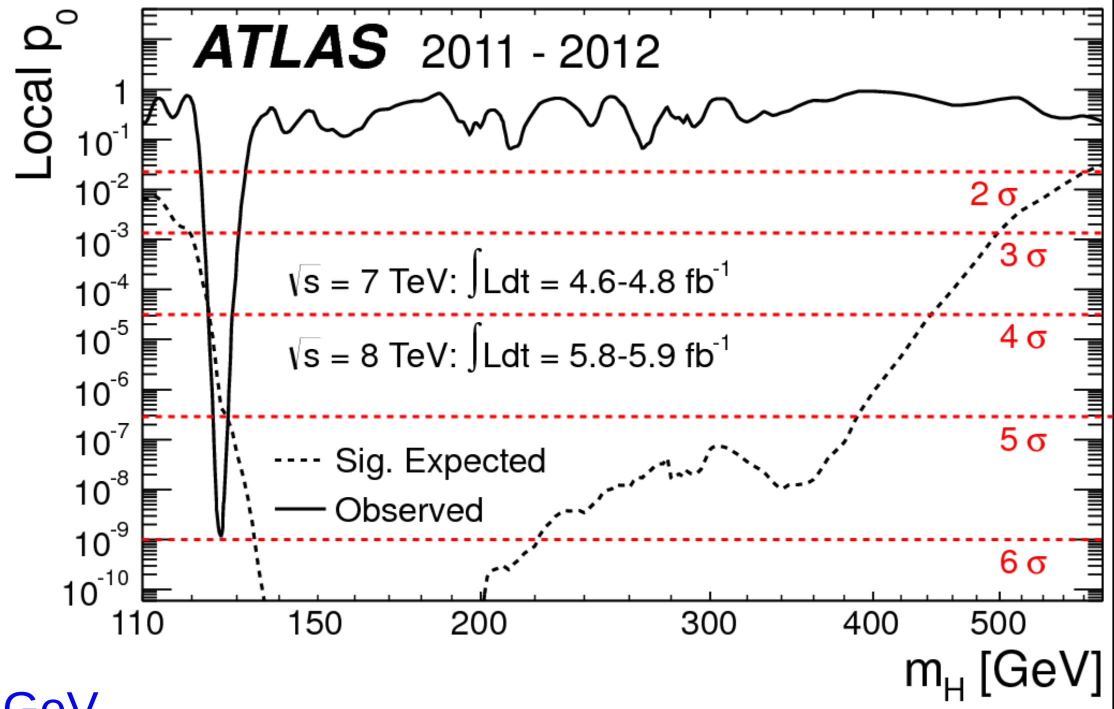
[1] 1-sided p-value is capped at $p_0 < 0.5$.
Can uncap by using $-Z_0$ for $\hat{\mu} < 0$

[2] $\Phi(Z)$ = ROOT::Math::gaussian_cdf(Z)
$\Phi^{-1}(p)$ = ROOT::Math::gaussian_quantile(p,1.0)

Discovery: p_0 vs m_H

PLB 716 (2012) 1

- Each mass hypothesis (m_H) has its own likelihood function, $L_{m_H}(\mu, \boldsymbol{\theta})$, eg.
  1. m_H hypothesis in kinematic fits
  2. $\mu = \sigma/\sigma_\mathrm{SM}(m_H)$ so need m_H-specific SM production XS and decay BR [LHC-H-XS-WG]
  3. each combined likelihood includes accessible decay modes at specified m_H



- p_0 vs m_H plot is the result of ~independent fits to each L_{m_H} [1]
  - The largest local significance is 6.0σ ($p_0 \sim 10^{-9}$) at $m_H = 126.5$ GeV
    - the result of many (part-correlated) searches across the full $110 \leq m_H < 600$ GeV range
  - correct for the "look-elsewhere effect" using Gross-Vitells formula [arXiv:1005.1891]:
    - $p_\mathrm{global} = p_\mathrm{local} + \langle N(c_0)\rangle e^{-(c-c_0)/2} \qquad = 10^{-9} + 9 \cdot e^{-6.0^2/2} = 1.4 \cdot 10^{-7} \ \rightarrow 5.1\sigma$

- Still using asymptotic approximation, which we may not be confident in for new signal
  → test with toys

[1] except in m_H measurement, use single likelihood $L(m_H, \boldsymbol{\theta})$

Discovery with toys

PRD 90 (2014) 052004

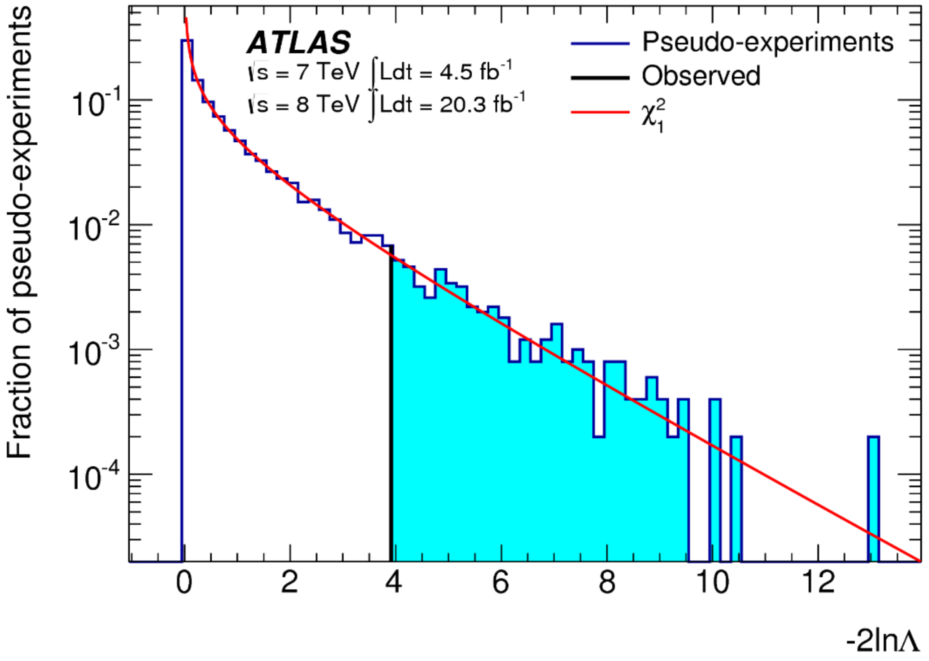
- Toy MC (AKA “Monte Carlo pseudo-experiments”) can be generated directly from the components of the likelihood function
  1. For each toy, generate
     1. toy dataset (pdf.generate(obs)), with $\mu, \boldsymbol{\theta}$ determined from expectation or fit to data
     2. set of global observables (pdf.generate(globObs))
        - simulates variation of “NP truth”
  2. Calculate a test statistic, $t_\mu = -2\ln\Lambda(\mu)$, requiring:
     1. conditional fit, under hypothesis being tested, eg. $\mu = 0$, background-only for p_0
     2. unconditional fit for best-fit $\hat{\mu}$ for this toy
  3. for signal significance, use one-sided capped-below profile likelihood ratio:

$$q_0 = \begin{cases} t_{\mu=0} & \text{if } \hat{\mu} > 0 \\ 0 & \text{if } \hat{\mu} \le 0 \end{cases}$$

- The observed p-value is just the fraction of toys with test statistic larger than the observed:
  - $p_0 = N_{\text{toys}}\left(q_0 > q_{0,\text{obs}}\right) / N_{\text{toys}}$

Example distribution of t_0

(here for a 2-sided test of compatibility of two signals, not 1-sided signal significance)

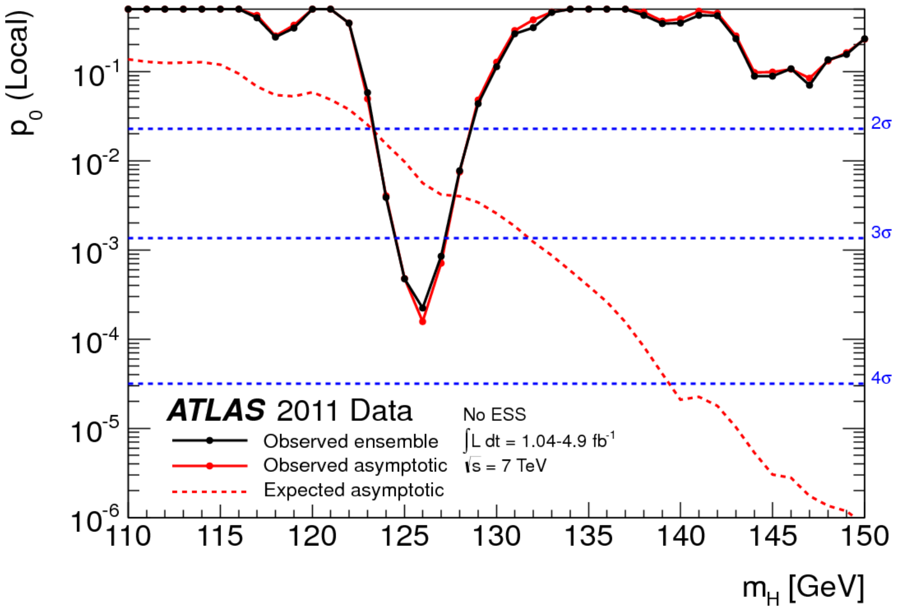


Discovery with toys

PLB 716 (2012) 1
PLB 710 (2012) 49

- For the 2012 ATLAS Higgs discovery
  - the 6.0σ local significance was reduced to 5.9σ by including the effect of energy-scale systematics
  - Significance of ESS could only be measured using toys at $m_H = 126.5$ GeV
    - limited by CPU time available (used extrapolation from 300k toys)

- The cross-check with toys is more clearly seen with a previous sample
  - lower significance → smaller number of toys required



Exclusion

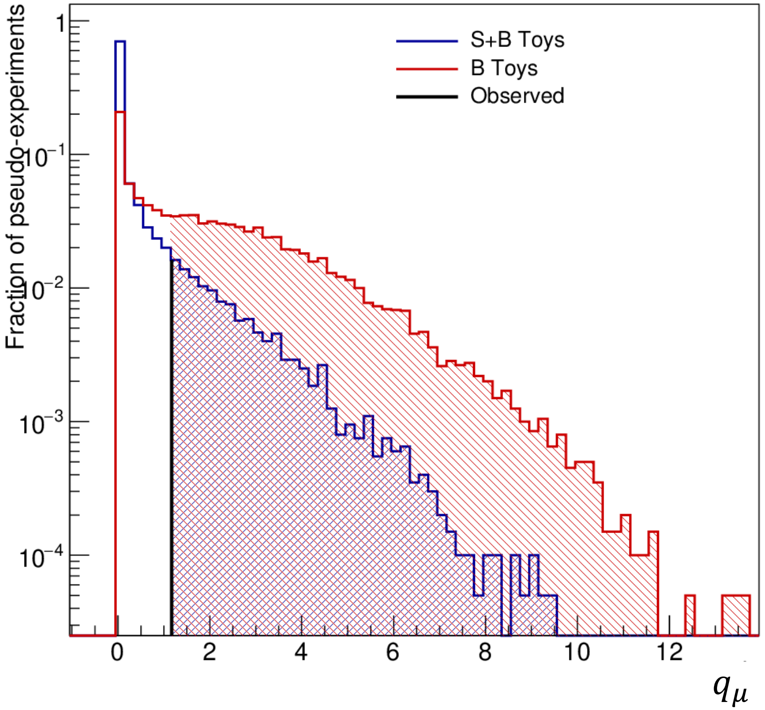
Exclusion: finding upper limit with CLs

Will will explain this in detail in the tutorial

- To set a limit the null-hypothesis is a particular signal+background hypothesis
  - here called $p_\mu = \mathrm{CL}_{s+b}$
- For an upper limit, we only want to exclude values below the limit
  - use test statistic: one-sided capped-above profile likelihood ratio

$$q_\mu = \begin{cases} t_\mu & \text{if } \hat{\mu} < \mu \\ 0 & \text{if } \hat{\mu} \geq \mu \end{cases}$$

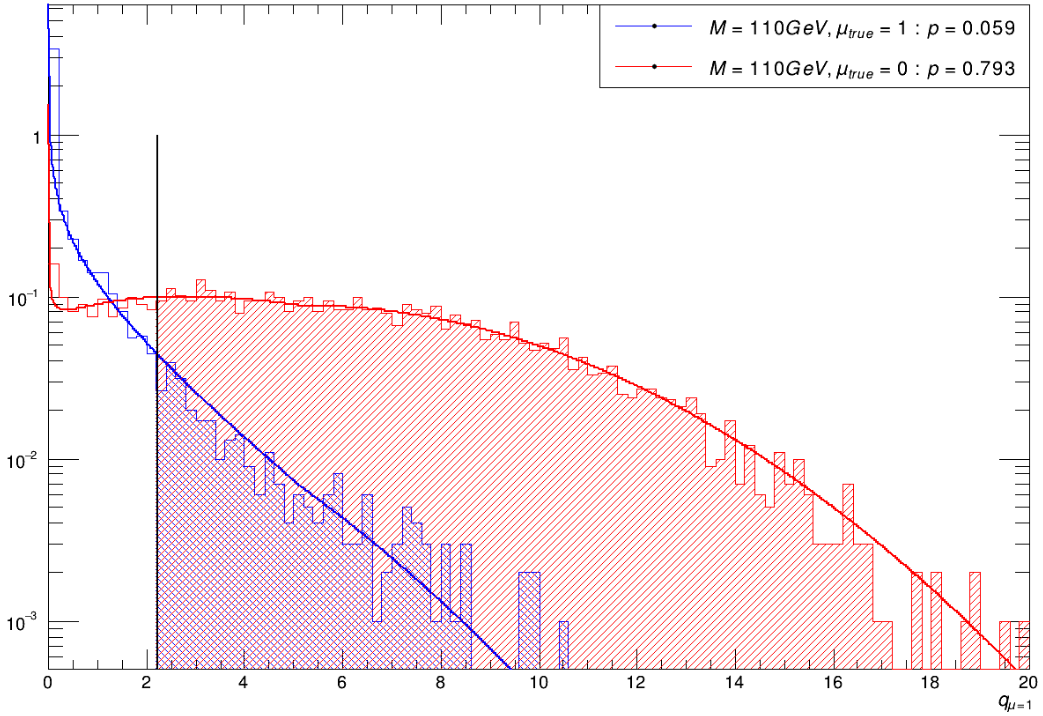
- In particle physics, we often use CLs instead of CL_{s+b} to set upper limits
  - CLs divides the tested p-value ($p_\mu = \mathrm{CL}_{s+b}$) by the background-exclusion p-value ($p_b = \mathrm{CL}_b$)
  - $p_{\mathrm{CL}s} = p_\mu / p_b$
    - with the expected background, $p_b = 0.5$, so this usually has little effect, but it is useful to inhibit a background fluctuation spuriously excluding a hypothesis to which we have little sensitivity
- p_μ and p_b can be estimated with toys similar to the procedure for discovery



Exclusion: CLs asymptotic approximation

Will will explain this in detail in the tutorial

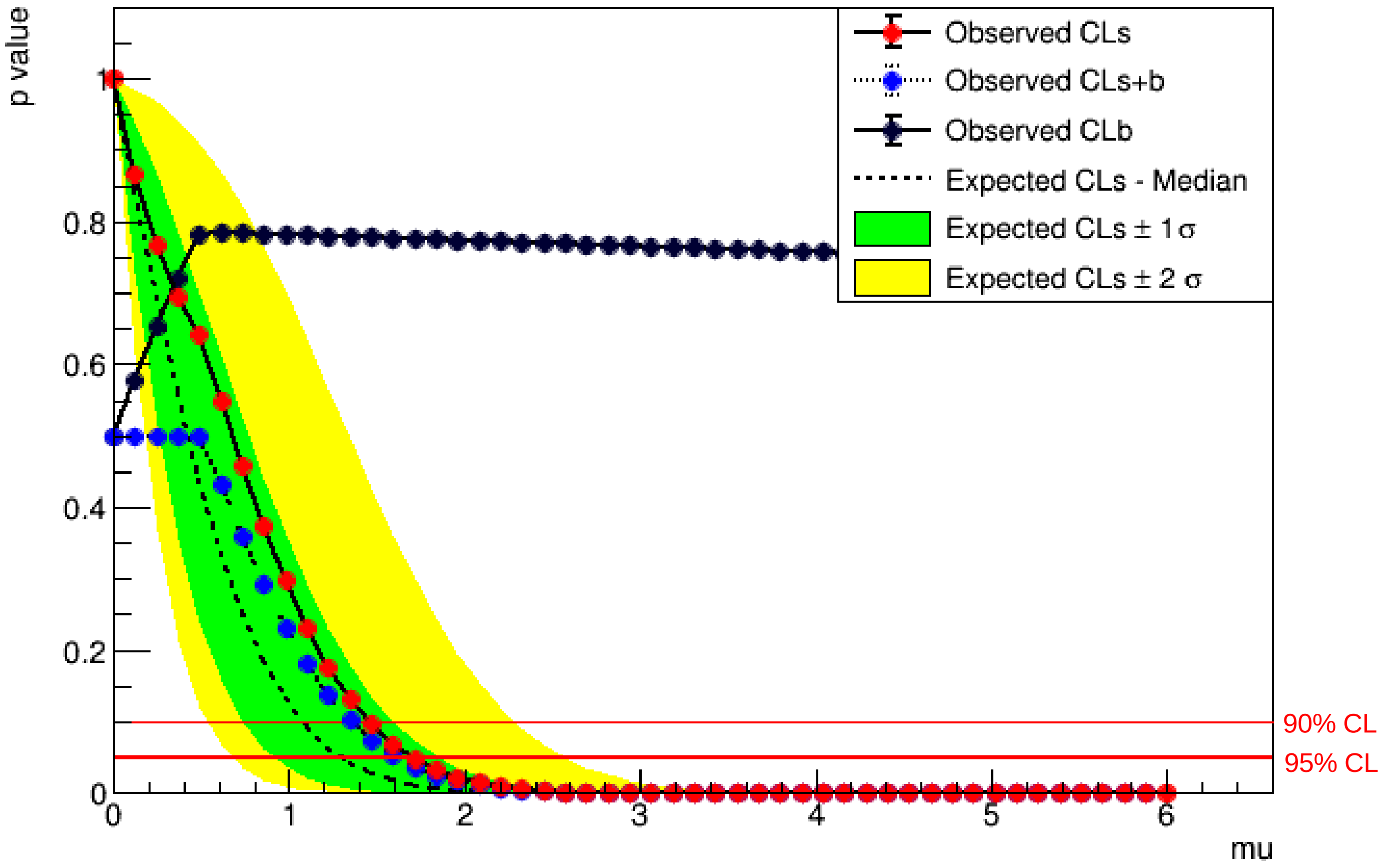
- Asymptotic limit obtained using the procedure from Asimov Paper [arXiv:1007.1727]
  - null hypothesis follows a χ^2 distribution with a δ-function at $q_\mu = 0$
  - alternative hypothesis follows a non-central χ^2 distribution
    - non-centrality parameter related to q_μ(Asimov)
- Various tools to calculate asymptotic p-values, eg.
  - RooStats::AsymptoticCalculator
  - provided with hands-on tutorial:



Exclusion limit setting with CLs

- For a 95% CL limit, reject a particular μ (s+b) hypothesis if $p_{\mathrm{CL}s} \le 0.05$.
  - to obtain a limit, find μ_{up}, the μ value for which $p_{\mathrm{CL}s} = 0.05$
- For toys, this means generating/fitting toys for various μ and interpolating μ_{up}
  - much faster to use asymptotic approximation
    - but may need to test validity using toys, eg. when only a few events

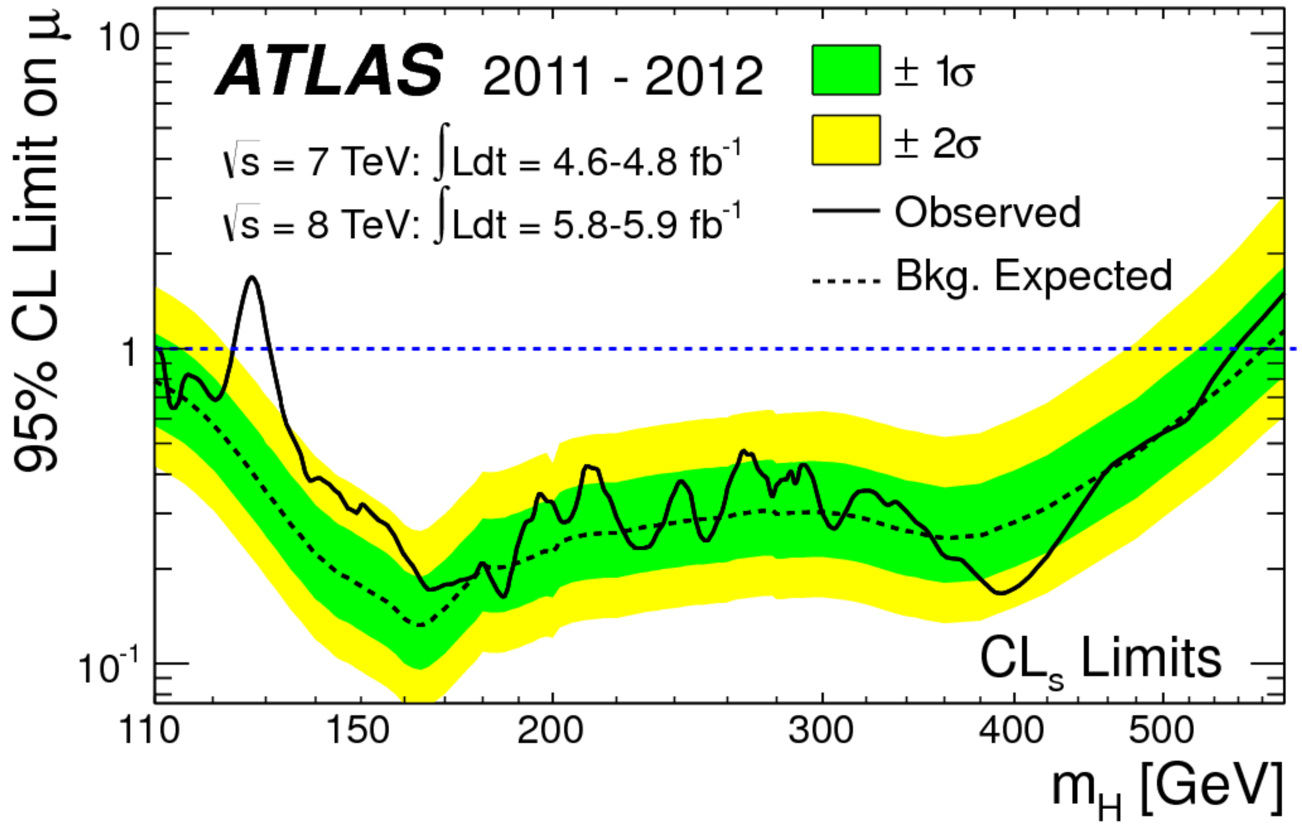
Will will explain this in detail in the tutorial



Exclusion limit vs m_H

PLB 716 (2012) 1

- In Higgs search, plot $\mu_{\rm up}$ vs m_H
  - different likelihood for each m_H, as before



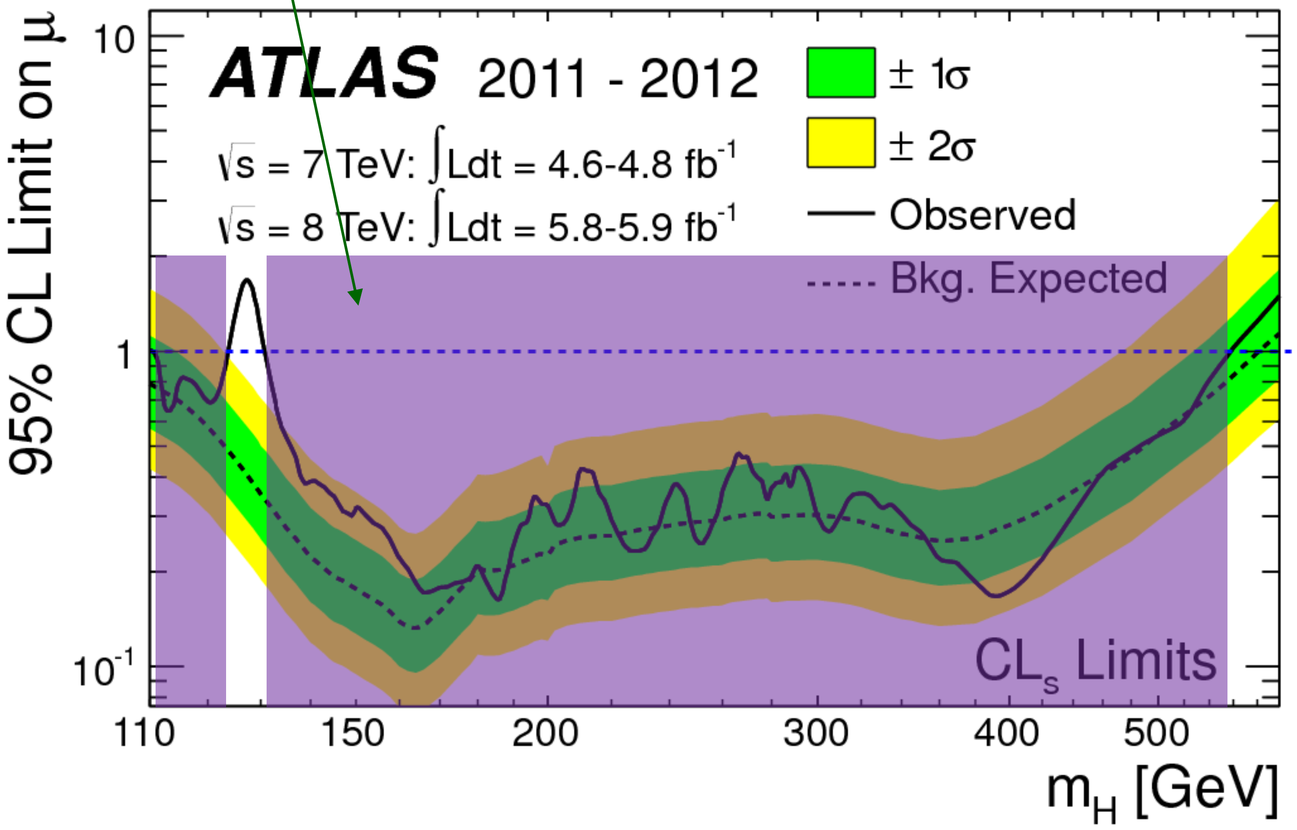
Exclusion limit vs m_H

PLB 716 (2012) 1

- In Higgs search, plot μ_{up} vs m_H
  - different likelihood for each m_H, as before

Observed

SM Higgs ($\mu = 1$) with these m_H excluded



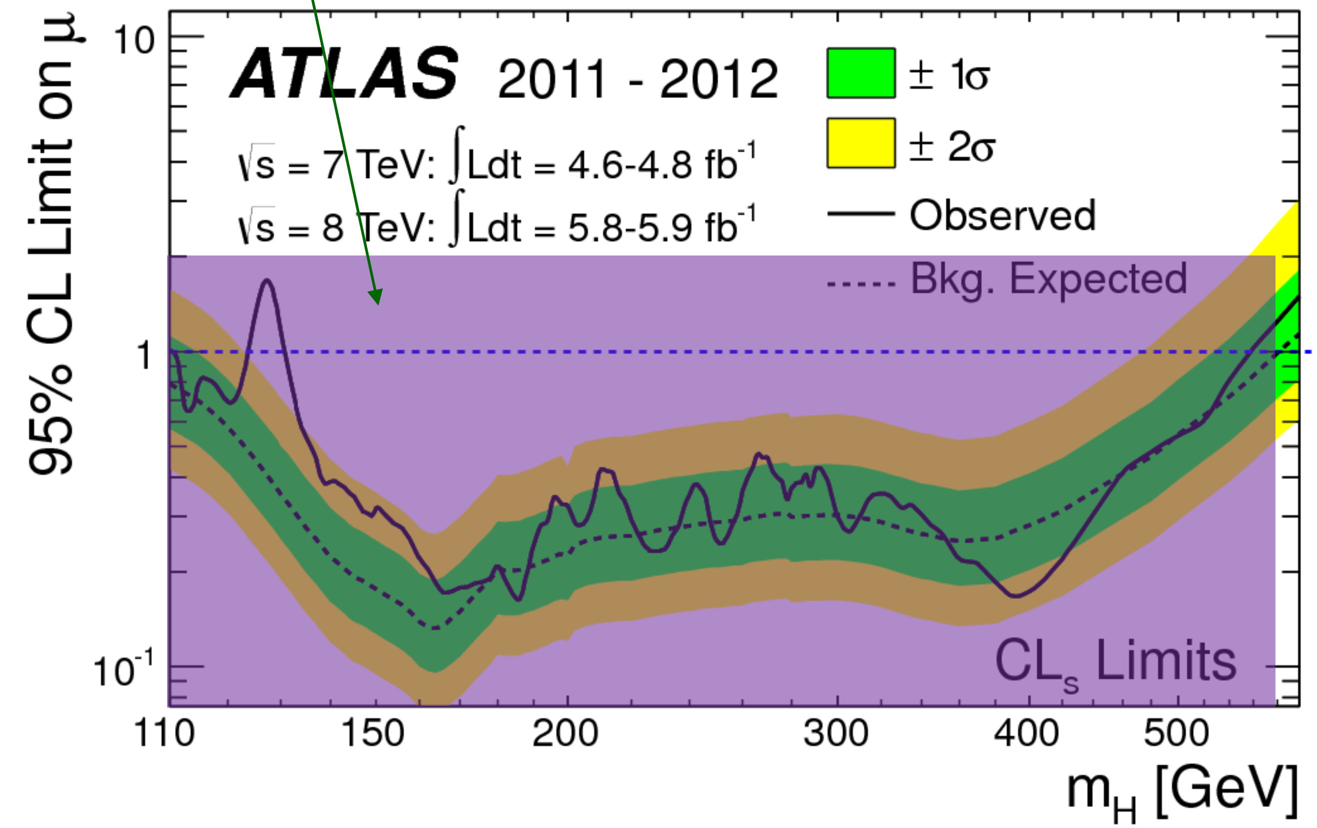
Exclusion limit vs m_H

PLB 716 (2012) 1

- In Higgs search, plot μ_{up} vs m_H
  - different likelihood for each m_H, as before

Expected

SM Higgs ($\mu = 1$) with these m_H excluded

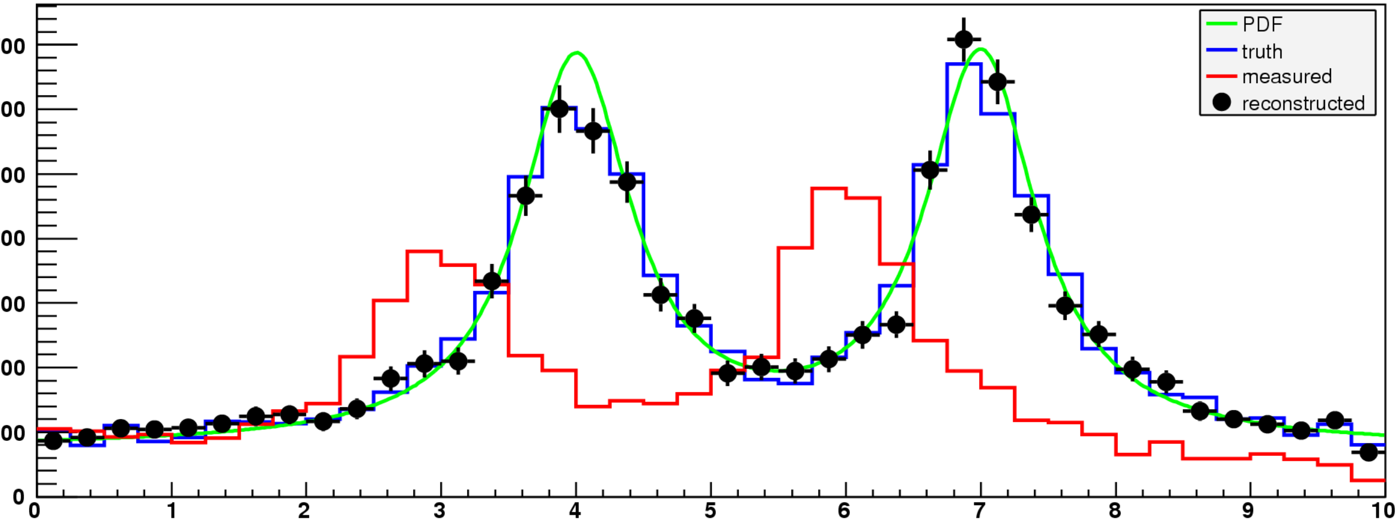


Unfolding

Unfolding – the problem

- In other fields known as “deconvolution” or “unsmearing”
  - often applied to “correct” images; we normally use it for histograms
- Given a “true” distribution, μ_j, that is corrupted by measurement/detector effects, described by a response matrix, R_{ij}, and background, β_i, we measure

$$\nu_i = \sum_{j=1}^{N} R_{ij}\mu_j + \beta_i \qquad \text{or} \qquad \boldsymbol{\nu} = \boldsymbol{R\mu} + \boldsymbol{\beta}$$



- This may involve
  1. bin migration and smearing
     - events moving between bins (off-diagonal R_{ij})
     - if not this, then don’t bother with unfolding
  2. inefficiencies
     - undetected events ($\epsilon_i = \sum_{j=1}^{N} R_{ij} < 1$)
  3. background / fake events
     - measured events not from true distribution ($\beta_i > 0$)

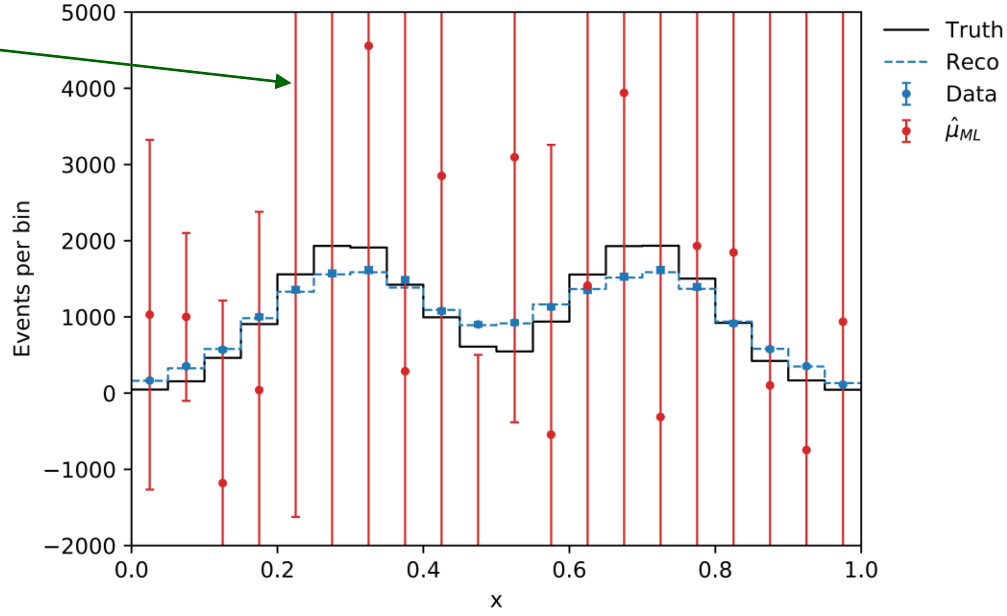
- We use unfolding to try to recover the true distribution

Regularisation

- We could unfold by inverting the response matrix

$$\boldsymbol{\nu} = \boldsymbol{R}^{-1}(\boldsymbol{\mu} - \boldsymbol{\beta})$$

- Statistical fluctuations mess this up: $\boldsymbol{R}^{-1}$ can't distinguish between a fluctuation and fine structure in the truth, $\boldsymbol{\nu}$
  - Results in large uncertainties in the unfolded result



Credit: Adam Bozson (RHUL, 2018)

- This can be resolved with some form of regularisation
  - reduce the statistical error by introducing a systematic bias
  - regularisation parameter controls degree of bias
  - many different methods available, e.g.
    - Simple correction factor method assumes similar bin-migrations as in MC
    - Tikhonov regularisation biases to a smooth distribution
    - Iterative Bayes unfolding biases towards initial guess (MC truth)

Unfolding advice

- Don’t unfold! (unless you can’t help it)
  - if you have the truth, apply (“fold”) the detector/measurement response to the truth
    - much better than trying to unfold the measurements back to the truth, e.g.
    1. if you have a functional model for your truth distribution, fit the model parameters
    2. if you have a theory prediction, compare the corrected predictions to your measurements
- Unfolding is needed if your final result has to be the “true” distribution, for which there is no model
  - e.g. for comparison with another experiment

- Don’t regularise if you don’t have to
  - If the statistical fluctuations are small compared to the bin migrations, can just invert
    - check the statistical uncertainties (especially bin-bin correlations) are acceptable

- If you have to regularise:
  - optimise regularisation parameter for bias vs statistical uncertainties
  - check bias with systematically independent MC samples (e.g. different event generators)
    - bias should be included in systematic uncertainties of the result
  - unfolding introduces bin-bin correlations – in bias and statistical errors
    - need to be understood and reported
  - ideally cross-check with more than one unfolding method

Unfolding software

- ROOT includes TSVDUnfold and TUnfold methods built-in

- RooUnfold package provides a common interface and tools for several unfolding methods:

1. unregularised matrix inversion
2. simple correction factors
3. Iterative Bayes unfolding (IBU)
4. SVD unfolding (via TSVDUnfold)
5. Tikhonov regularization with least square fit (via TUnfold)
6. unfolding with Gaussian Processes (GP)
7. Iterative dynamically-stabilized unfolding (IDS)

Summary

Summary of model building

$$L(\boldsymbol{\mu},\boldsymbol{\theta}) = \prod_c \prod_i P_c(x_i|\boldsymbol{\mu},\boldsymbol{\theta}) \cdot \prod_j C_j(g_j|\theta_j)$$

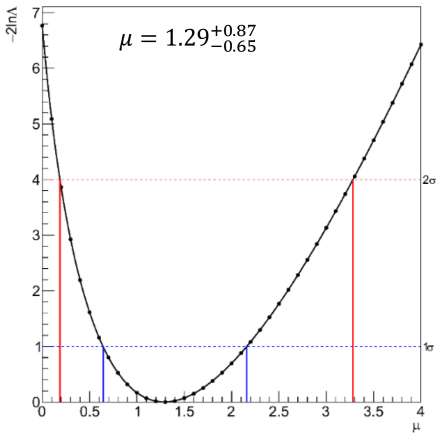
- Model PDF, function of
  - observables
  - parameters of interest (POIs)
  - nuisance parameters (NPs)
- Dataset
  - Entries containing values of some of the observables
  - Global observables are common to all entries
- Likelihood fit minimise $-2\ln L$

- Build models with
  1. RooFit (C++, Python, or factory)
  2. HistFactory (XML)
  3. pyhf (JSON)
  4. zfit (Python)
- Keep model and data in RooFit workspace files
- Asimov dataset allows tests of the model expectation

Summary of statistical tests

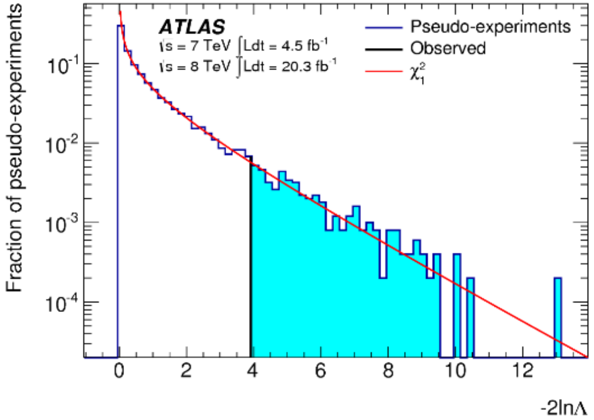
- Measurement, scanning profile likelihood ratio
  - tools: RooFit
  - test statistic: (two-sided) profile likelihood ratio

$$\mathrm{t}_\mu = -2\ln\Lambda(\mu) = -2\ln\frac{L\left(\mu,\hat{\hat{\boldsymbol{\theta}}}(\mu)\right)}{L(\hat{\mu},\hat{\boldsymbol{\theta}})}$$



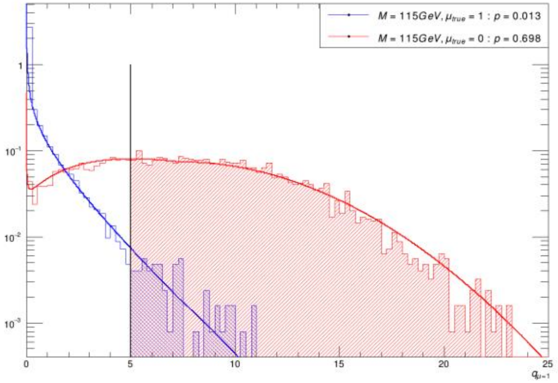
- Discovery with profile likelihood ratio, asymptotic or toys
  - tools: RooFit, RooStats
  - test statistic: one-sided capped-below profile likelihood ratio

$$q_0 = \begin{cases} t_{\mu=0} & \text{if } \hat{\mu} > 0 \\ 0 & \text{if } \hat{\mu} \leq 0 \end{cases}$$



- Exclusion with CLs, asymptotic or toys
  - tools: RooFit, RooStats, or pyhf
  - test statistic: one-sided capped-above profile likelihood ratio

$$q_\mu = \begin{cases} t_\mu & \text{if } \hat{\mu} < \mu \\ 0 & \text{if } \hat{\mu} \geq \mu \end{cases}$$



Backup

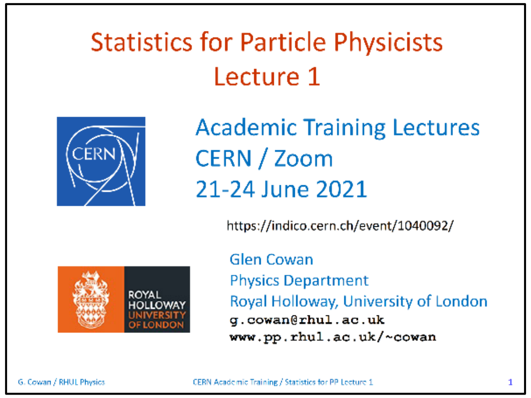
More help

- For a more thorough introduction, I recommend:

1. CERN Academic Training Lecture series, which has had 3–4 hour lectures by different HEP statistics experts every couple of years.
   a. “Statistics for Particle Physicists”, by Glen Cowan (June 2021)

   previous lectures also useful, e.g.

   b. Eilam Gross in 2018
   c. Glen Cowan in 2012 (part 4 on unfolding)
   d. Kyle Cranmer in 2011



Statistics for Particle Physicists
Lecture 1

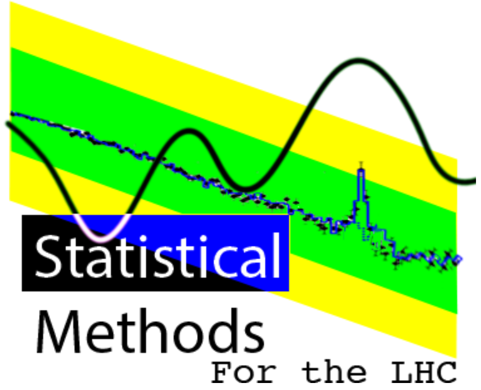
Academic Training Lectures
CERN / Zoom
21-24 June 2021

https://indico.cern.ch/event/1040092/

Glen Cowan
Physics Department
Royal Holloway, University of London
g.cowan@rhul.ac.uk
www.pp.rhul.ac.uk/~cowan

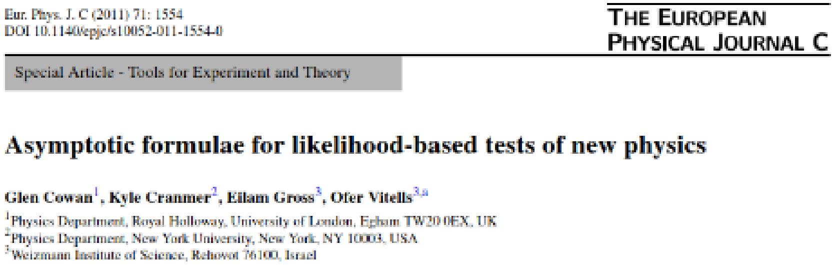
G. Cowan / RHUL Physics CERN Academic Training / Statistics for PP Lecture 1 1

2. “Statistics Methods for the LHC” – online documentation from ATLAS, with RooFit / RooStats / RooUnfold code examples.



3. “Asymptotic formulae for likelihood-based tests of new physics”

Cowan, Cranmer, Gross, Vitells, arXiv:1007.1727, EPJ C (2011) 71:1554



Eur. Phys. J. C (2011) 71: 1554
DOI 10.1140/epjc/s10052-011-1554-0

THE EUROPEAN PHYSICAL JOURNAL C

Special Article - Tools for Experiment and Theory

Asymptotic formulae for likelihood-based tests of new physics

Glen Cowan[1], Kyle Cranmer[2], Eilam Gross[3], Ofer Vitells[3,a]

[1] Physics Department, Royal Holloway, University of London, Egham TW20 0EX, UK
[2] Physics Department, New York University, New York, NY 10003, USA
[3] Weizmann Institute of Science, Rehovot 76100, Israel

RooStats, HistFactory, and pyhf

1. RooStats (ROOT built-in) provides higher-level statistical analysis tools
   - eg. ProfileLikelihoodTestStat, AsymptoticCalculator, FrequentistCalculator, HypoTestInverter
2. HistFactory (ROOT built-in) is a tool for creating models of binned data with systematics

$$L(\boldsymbol{\mu},\boldsymbol{\theta}) = \prod_c \prod_i \text{Poisson}(n_i|\nu_i(\boldsymbol{\mu},\boldsymbol{\theta})) \cdot \prod_j \text{Gaussian}(g_j|\theta_j,\sigma_j)$$

   - Multiple disjoint channels, multiple samples contributing to each with additional (possibly shared) systematics
   - Many analyses can use HistFactory instead of calling RooFit directly.
   - Model specified with XML, which refers to histograms in hist.root files

These tools can be used from Python, but non-ROOT Python alternatives are also available:

1. pyhf is a reimplementation of HistFactory in pure-Python
   - no dependence on ROOT or RooFit
   - XML+histograms specification replaced by JSON
     - JSON is easier to read and modify
   - full conversion of models from HistFactory and back
   - reproduces HistFactory results [tested]
     - pyhf allows other minimisation techniques, not just MINUIT (CERN, 1975–)
       - supports multi-threading and GPUs
       - so far, for most HEP applications, RooFit / MINUIT is just as fast
2. zfit is a general-purpose model fitting package, using TensorFlow

Hypothesis Tests

- Exclusion and Discovery plots present the results of a collection of Hypothesis Tests
  - A Hypothesis Test is really the process of calculating a p-value and seeing whether its less than or greater than a critical value (0.05 in the case of 95% CL)
- Hypothesis Space: parameters of the signal model we want to study (parameter grid)
- Test Statistic to perform hypothesis tests with
  - Exclusions: one-sided capped-above Profile Likelihood Ratio Test Statistic q_μ
  - Discovery: one-sided capped-below Profile Likelihood Ratio Test Statistic q_0
- Types of p-values:
  - null p-value: The p-value under the null hypothesis (the hypothesis being tested)
    - In exclusion tests the null hypothesis is a particular s+b hypothesis (CL$_{s+b}$)
    - In discovery tests the null hypothesis is the background-only hypothesis (p_0)
  - alternative p-value: The p-value under an alternative hypothesis
    - only relevant for exclusions (also called CL$_b$)
  - CLs p-value: The ratio of the above two p-values
- Type of measurement:
  - Observed p-value / limit, based on event data
  - Expected p-value / limit, based on a particular model
    - eg. SM, background only, signal model
    - often shown with median line and $\pm 1\sigma$, $\pm 2\sigma$ bands

Exclusion: asymptotic CLs

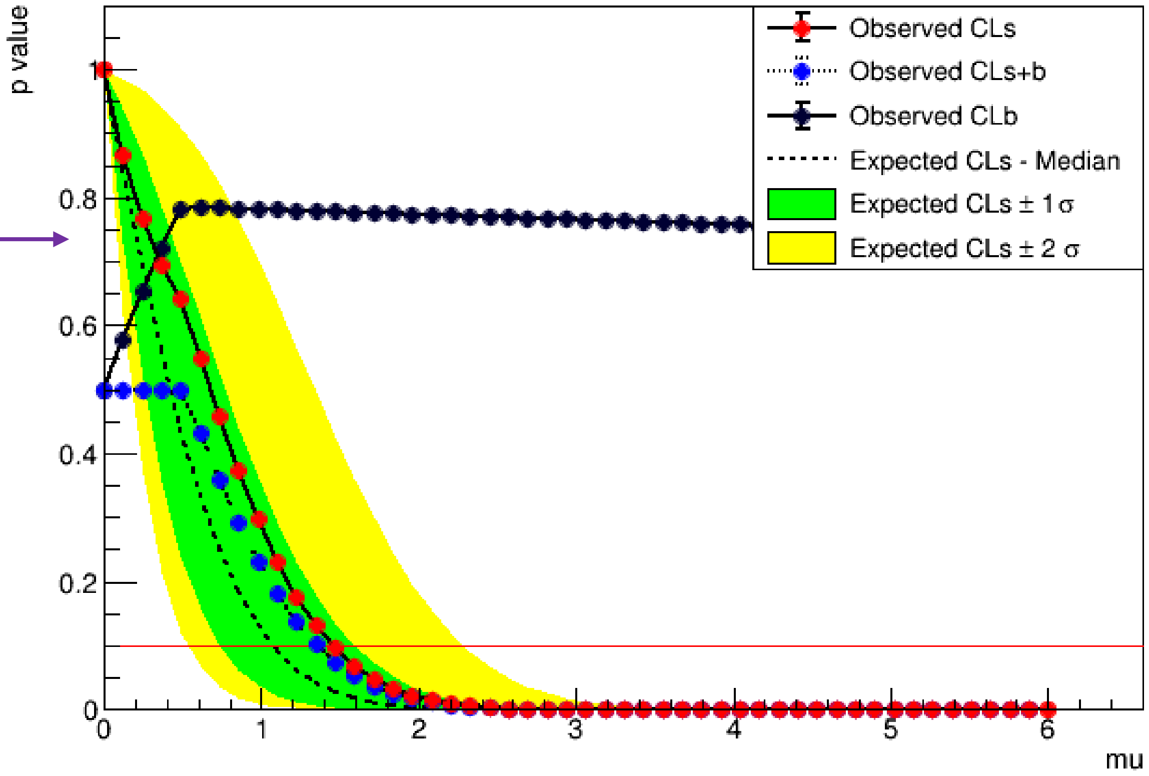
- CLs: $p_{\mathrm{CL}s} = p_\mu / p_b$
  - CLs divides the tested p-value (CL_{s+b}) by the background-exclusion p-value (CL_b)
    - normally has little effect, but it is useful to inhibit a fluctuation spuriously excluding a hypothesis to which we have little sensitivity
- For a 95% CL limit, reject a particular μ (s+b) hypothesis if $p_{\mathrm{CL}s} \leq 0.05$.
  - to obtain a limit, find μ_{up}, the μ value for which $p_{\mathrm{CL}s} = 0.05$

- Asymptotic limit obtained using the procedure from Asimov Paper [arXiv:1007.1727]
  - $q_\mu \quad = -2\ln\Lambda(\mu)$ PLR for observed data
  - $q_{\mu,A} = -2\ln\Lambda_A(\mu|0)$ PLR for background-only Asimov dataset
  - $p_{\mathrm{CL}s} = (1 - \Phi(\sqrt{q_\mu}\,))$ / $\Phi(\sqrt{q_{\mu,A}} - \sqrt{q_\mu})$
    - scan μ to find μ_{up} for which $p_{\mathrm{CL}s} = 0.05$.
  - For the median expected limit, $\mu_{\mathrm{up}} = 1.96\,\sigma(\mu_{\mathrm{up}})$ $[\Phi^{-1}(1 - 0.05/2) = 1.96]$
    - where $\sigma(\mu_{\mathrm{up}}) = \mu_{\mathrm{up}}/\sqrt{q_{\mu_{\mathrm{up}},A}}$, so again requires a numerical determination of μ_{up}
    - The expected bands, median$\pm N\sigma$, $\mu_{\mathrm{up}+N} = \left(\Phi^{-1}\left(1 - 0.05\Phi(N)\right) + N\right) \cdot \sigma(\mu_{\mathrm{up}+N})$

CLs procedure with RooStats

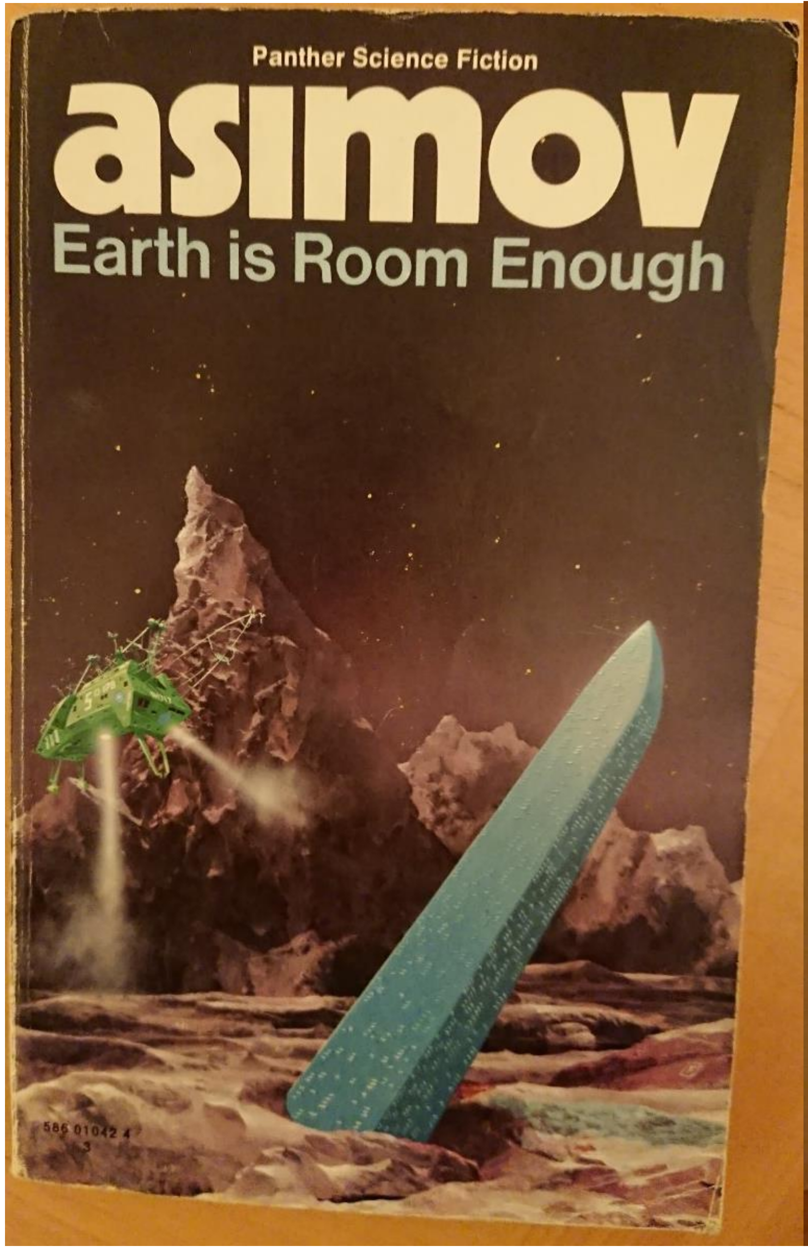
- For RooFit models, see:
  1. RooStats StandardHypoTestInvDemo.C tutorial, or
  2. ATLAS CLs tutorial
- In summary, create an asymptotic or toy calculator:
  1. RooStats::AsymptoticCalculator calc (data, bModel, sbModel);   // or
  2. RooStats::FrequentistCalculator calc (data, bModel, sbModel);
- and pass that to the hypothesis test inverter:

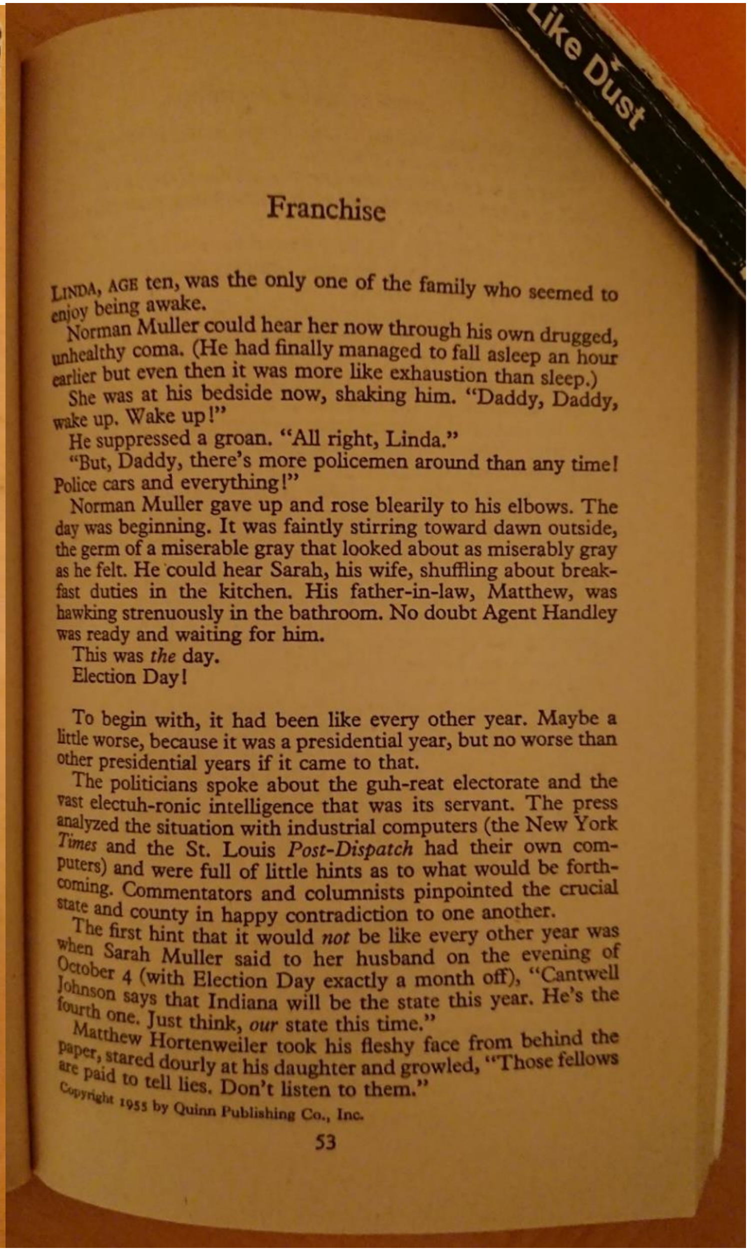
```
RooStats::HypoTestInverter  hypo (calc);
result = hypo.GetInterval();
RooStats::HypoTestInverterPlot (,,result);
```



- For HistFactory-style models,
  pyhf has built-in tools to calculate CLs

Isaac Asimov – Franchise





Franchise

LINDA, AGE ten, was the only one of the family who seemed to enjoy being awake.

Norman Muller could hear her now through his own drugged, unhealthy coma. (He had finally managed to fall asleep an hour earlier but even then it was more like exhaustion than sleep.)

She was at his bedside now, shaking him. "Daddy, Daddy, wake up. Wake up!"

He suppressed a groan. "All right, Linda."

"But, Daddy, there's more policemen around than any time! Police cars and everything!"

Norman Muller gave up and rose blearily to his elbows. The day was beginning. It was faintly stirring toward dawn outside, the germ of a miserable gray that looked about as miserably gray as he felt. He could hear Sarah, his wife, shuffling about breakfast duties in the kitchen. His father-in-law, Matthew, was hawking strenuously in the bathroom. No doubt Agent Handley was ready and waiting for him.

This was *the* day.

Election Day!

To begin with, it had been like every other year. Maybe a little worse, because it was a presidential year, but no worse than other presidential years if it came to that.

The politicians spoke about the guh-reat electorate and the vast electuh-ronic intelligence that was its servant. The press analyzed the situation with industrial computers (the New York *Times* and the St. Louis *Post-Dispatch* had their own computers) and were full of little hints as to what would be forthcoming. Commentators and columnists pinpointed the crucial state and county in happy contradiction to one another.

The first hint that it would *not* be like every other year was when Sarah Muller said to her husband on the evening of October 4 (with Election Day exactly a month off), "Cantwell Johnson says that Indiana will be the state this year. He's the fourth one. Just think, *our* state this time."

Matthew Hortenweiler took his fleshy face from behind the paper, stared dourly at his daughter and growled, "Those fellows are paid to tell lies. Don't listen to them."

53

The Asimov dataset [arXiv:1007.1727] is named for SF author, Isaac Asimov, whose 1955 short story, *Franchise*, envisaged the 2008 US Presidential Election decided by one voter representative of the entire electorate.

This is my copy of the story, in a collection.

Statistics tutorial this afternoon

- The hands-on tutorial session uses a minimal amount of RooFit, but it is important that you have some familiarity with the following important RooFit classes:
  - Variables: RooRealVar, RooCategory
  - Collections: RooArgSet, RooArgList
  - Datasets: RooDataSet
  - PDFs: RooAbsPdf
  - Fit Results: RooFitResult
  - Workspaces: RooWorkspace
- If you are familiar with working with all of these classes then you are ready for the hands-on tutorial!
  - For everyone else, we have some materials and exercises for you to go through ahead of the session, which will make you familiar enough with these objects for the session.

- Instructions:
  - Login to monty.stfc.ac.uk
    - if you haven't got a login, please contact Will, will.buttinger@stfc.ac.uk
  - Clone the materials:

    ```
    git clone https://gitlab.cern.ch/will/ralstats.git
    ```

    and follow the Prerequisite.ipynb notebook.

- The material should take 1 to 2 hours to go through if you have no prior RooFit knowledge.