

RooFit, RooStats – Building, debugging & analyzing
statistical models
with special focus on profile likelihood

W. Verkerke (Nikhef & University of Amsterdam)

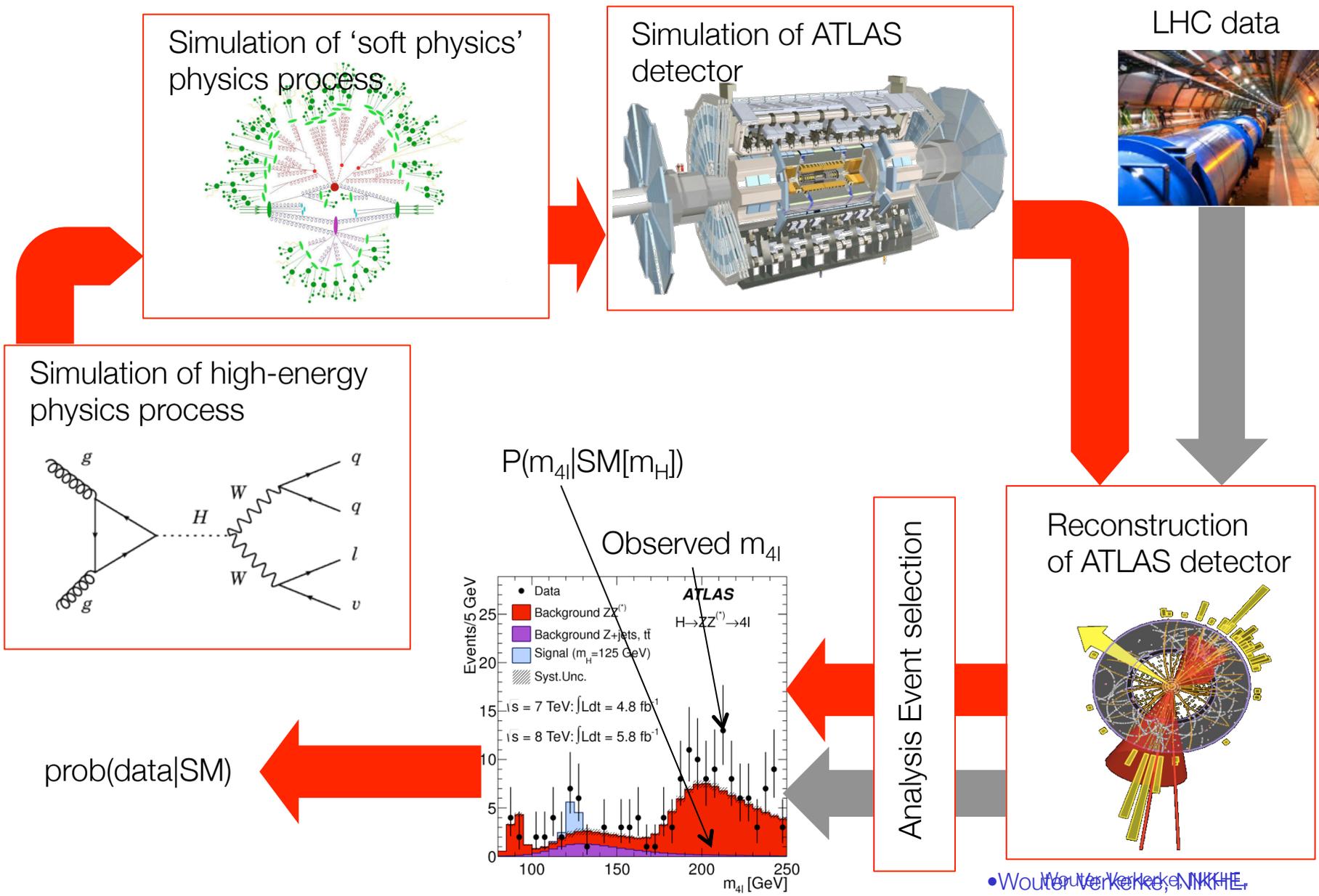
Roadmap of today's course

- Software 1 – Overview: building and analyzing models with RooFit, RooStats & HistFactory
- Modeling 1 – Systematic uncertainties, “Profile Likelihood”: Counting models with nuisance parameters
- Statistics 1 – Treatment of nuisance parameters in statistics inference
- Software 2 – Hands-on: Counting models with NPs in RooFit limits & confidence intervals with RooStats
- Modeling 2 – Modeling distributions with nuisance parameters, basics of template morphing
- Modeling 3 – Understanding complex fits – debugging techniques
- Software 3 – Hands-on: Modeling (un)binned distributions in RooFit, Combination & reparametrization

Software 1

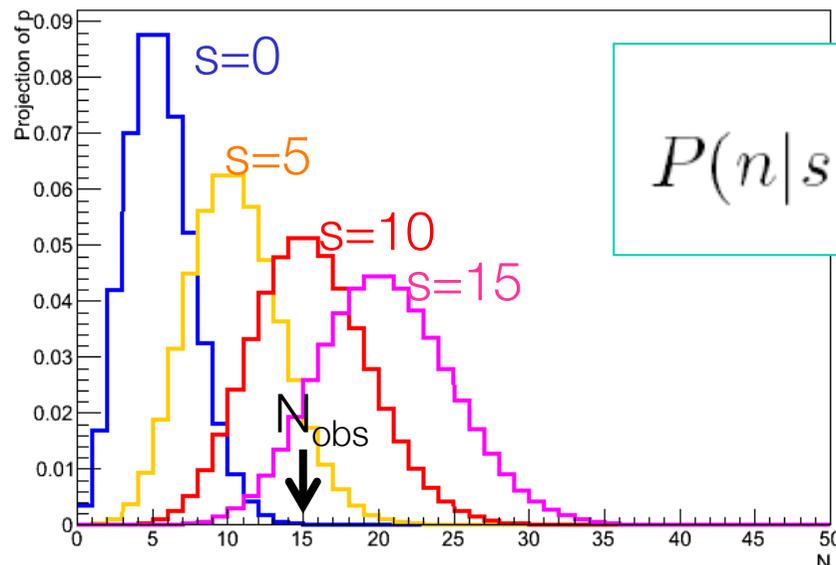
**Overview:
building and analyzing models
with
RooFit, RooStats & HistFactory**

The HEP analysis workflow illustrated



All experimental results start with the formulation of a model

- Examples of HEP **physics** models being tested
 - SM with $m(\text{top})=172,173,174$ GeV \rightarrow Measurement top quark mass
 - SM with/without Higgs boson \rightarrow Discovery of Higgs boson
 - SM with composite fermions/Higgs \rightarrow Measurement of Higgs coupling properties
- Via chain of physics simulation, showering MC, detector simulation and analysis software, a physics model is reduced to a statistical model
- A **statistical** model defines $p(\text{data}|\text{theory})$ for all observable outcomes
 - Example of a statistical model for a counting measurement with a known background



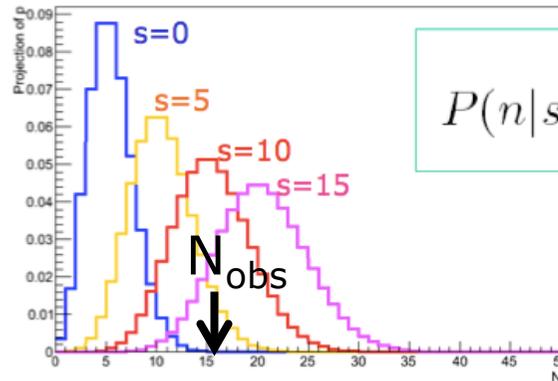
$$P(n|s + b) = \frac{(s + b)^n}{n!} e^{-(s+b)}$$

NB: b is a constant in this example

Definition: the Likelihood is $P(\text{observed data}|\text{theory})$

Everything starts with the likelihood

- **All** fundamental statistical procedures are based on the likelihood function as ‘description of the measurement’

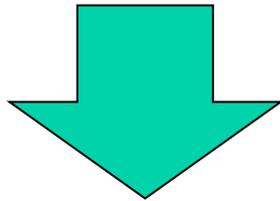


$$P(n|s+b) = \frac{(s+b)^n}{n!} e^{-(s+b)}$$

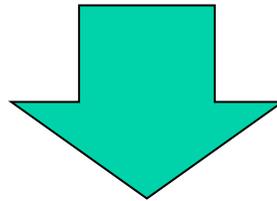
NB: b is a constant in this example

Definition: the Likelihood is $P(\text{observed data}|\text{theory})$

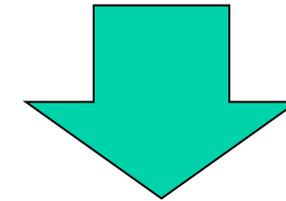
e.g. $L(15|s=0)$
e.g. $L(15|s=10)$



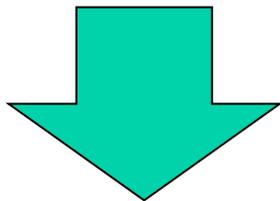
Frequentist statistics



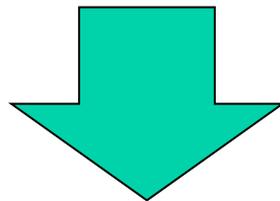
Bayesian statistics



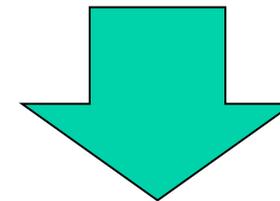
Maximum Likelihood



Confidence interval on s



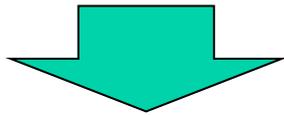
Posterior on s



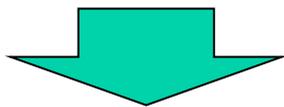
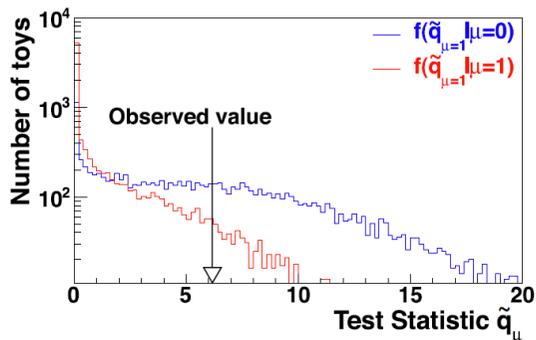
$s = x \pm y$

Everything starts with the likelihood

Frequentist statistics

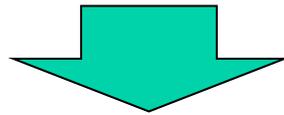


$$\lambda_{\mu}(\vec{N}_{obs}) = \frac{L(\vec{N} | \mu)}{L(\vec{N} | \hat{\mu})}$$

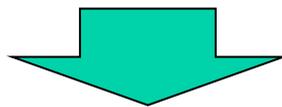
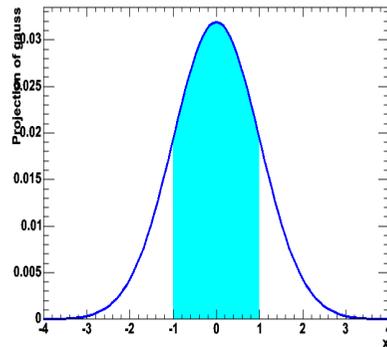


**Confidence interval
or p-value**

Bayesian statistics

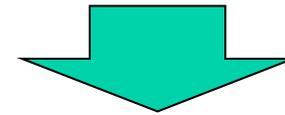


$$P(\mu) \propto L(x | \mu) \cdot \pi(\mu)$$

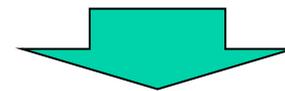
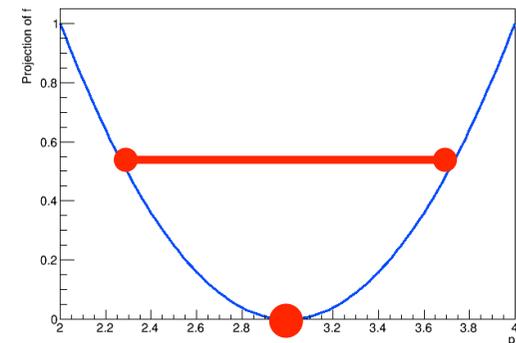


**Posterior on s
or Bayes factor**

Maximum Likelihood



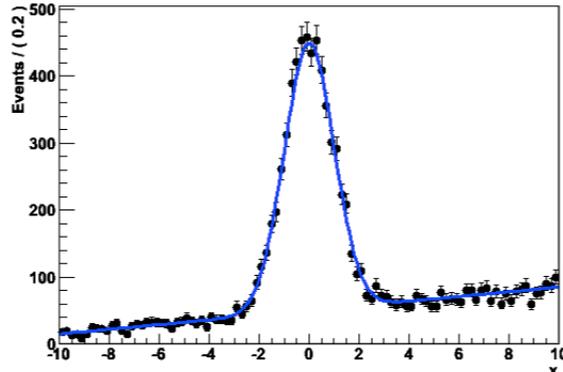
$$\left. \frac{d \ln L(\vec{p})}{d \vec{p}} \right|_{p_i = \hat{p}_i} = 0$$



s = x ± y

How is Higgs discovery different from a simple fit?

Gaussian + polynomial



ROOT TH1

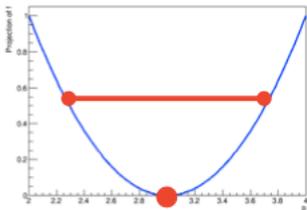
ROOT TF1

$$L(\vec{N} | \mu, \vec{\theta}) = \prod_i \text{Poisson}(N_i | f(x_i, \mu, \vec{\theta}))$$

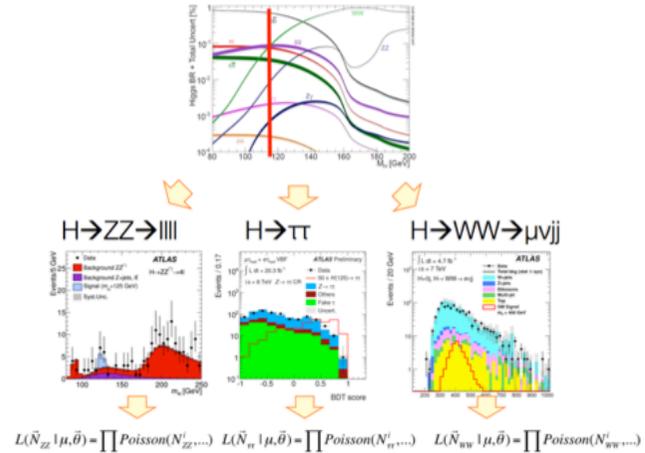
"inside ROOT"

ML estimation of parameters μ, θ using MINUIT (MIGRAD, HESSE, MINOS)

$$\mu = 5.3 \pm 1.7$$



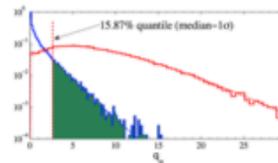
Higgs combination model



$$L(\vec{N}_{ZZ} | \mu, \vec{\theta}) = \prod \text{Poisson}(N'_{ZZ}, \dots) \quad L(\vec{N}_{\tau\tau} | \mu, \vec{\theta}) = \prod \text{Poisson}(N'_{\tau\tau}, \dots) \quad L(\vec{N}_{WW} | \mu, \vec{\theta}) = \prod \text{Poisson}(N'_{WW}, \dots)$$

$$L(\vec{N}_{ZZ}, \vec{N}_{\tau\tau}, \vec{N}_{WW} | \mu, \vec{\theta}) = \prod \text{Poisson}(N'_{ZZ}, \dots) \cdot \prod \text{Poisson}(N'_{\tau\tau}, \dots) \cdot \prod \text{Poisson}(N'_{WW}, \dots) \dots$$

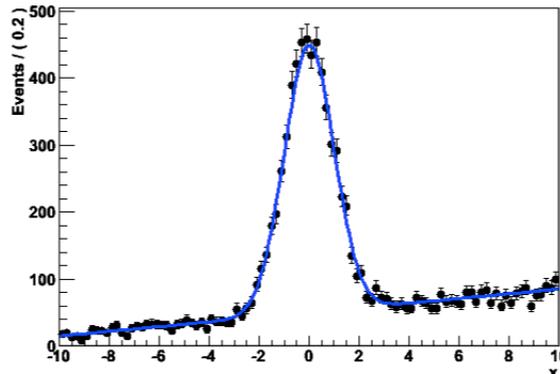
$$\lambda_{\mu}(\vec{N}_{ZZ}, \vec{N}_{WW}, \vec{N}_{\tau\tau}) = \frac{L(\vec{N}_{ZZ}, \vec{N}_{WW}, \vec{N}_{\tau\tau} | \mu, \hat{\theta})}{L(\vec{N}_{ZZ}, \vec{N}_{WW}, \vec{N}_{\tau\tau} | \hat{\mu}, \hat{\theta})}$$



$$p(H_{\mu}) = \int_{\lambda_{obs}}^{\infty} f(\lambda | H_{\mu}) d\lambda = \dots$$

How is Higgs discovery different from a simple fit?

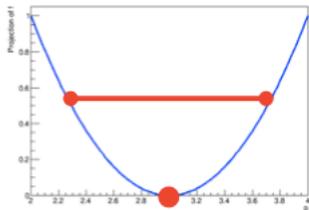
Gaussian + polynomial



ROOT TH1 ROOT TF1

$$L(\vec{N} | \mu, \vec{\theta}) = \prod_i \text{Poisson}(N_i | f(x_i, \mu, \vec{\theta}))$$

“inside ROOT”



ML estimation of parameters μ, θ using MINUIT (MIGRAD, HESSE, MINOS)

$$\mu = 5.3 \pm 1.7$$

Likelihood Model
orders of magnitude more complicated. Describes

- O(100) signal distributions
- O(100) control sample distr.
- O(1000) parameters representing syst. uncertainties

$$L(\vec{N}_{ZZ}, \vec{N}_{\tau\tau}, \vec{N}_{WW} | \mu, \vec{\theta}) = \prod \text{Poisson}(N_{ZZ}^i | \dots) \cdot \prod \text{Poisson}(N_{\tau\tau}^i | \dots) \cdot \prod \text{Poisson}(N_{WW}^i | \dots) \dots$$

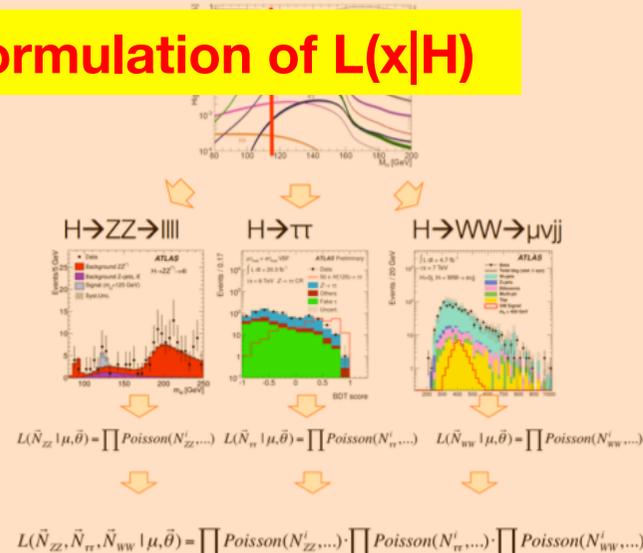
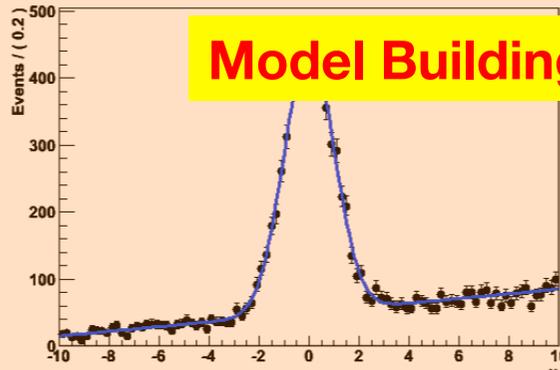
Frequentist confidence interval construction and/or p-value calculation not available as ‘ready-to-run’ algorithm in ROOT

How is Higgs discovery different from a simple fit?

Gaussian + polynomial

Higgs combination model

Model Building phase (formulation of $L(x|H)$)



ROOT TH1

ROOT TF1

$$L(\vec{N} | \mu, \vec{\theta}) = \prod_i \text{Poisson}(N_i | f(x_i, \mu, \vec{\theta}))$$

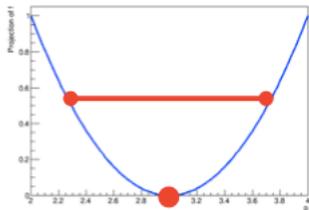
"inside ROOT"



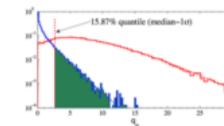
ML estimation of parameters μ, θ using MINUIT (MIGRAD, HESSE, MINOS)



$$\mu = 5.3 \pm 1.7$$



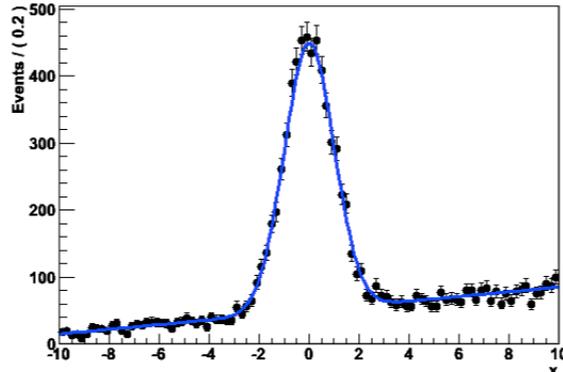
$$\lambda_{\mu}(\vec{N}_{ZZ}, \vec{N}_{WW}, \vec{N}_{\tau}) = \frac{L(\vec{N}_{ZZ}, \vec{N}_{WW}, \vec{N}_{\tau} | \mu, \hat{\theta})}{L(\vec{N}_{ZZ}, \vec{N}_{WW}, \vec{N}_{\tau} | \hat{\mu}, \hat{\theta})}$$



$$p(H_{\mu}) = \int_{\lambda_{obs}}^{\infty} f(\lambda | H_{\mu}) d\lambda = \dots$$

How is Higgs discovery different from a simple fit?

Gaussian + polynomial



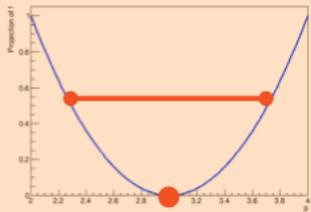
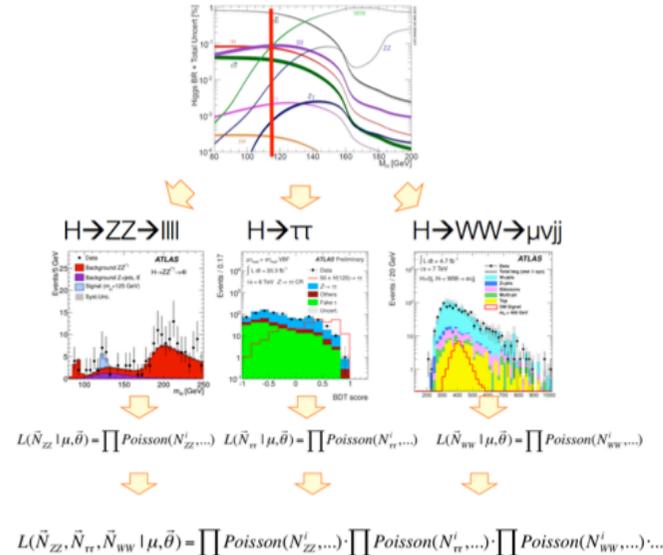
ROOT TH1

ROOT TF1

$$L(\vec{N} | \mu, \vec{\theta}) = \prod_i \text{Poisson}(N_i | f(x_i, \mu, \vec{\theta}))$$

“inside ROOT”

Higgs combination model



ML estimation of parameters μ, θ using MINUIT (MIGRAD, HESSE, MINOS)

$$\lambda_{\mu}(\vec{N}_{ZZ}, \vec{N}_{WW}, \vec{N}_{\tau\tau}) = \frac{L(\vec{N}_{ZZ}, \vec{N}_{WW}, \vec{N}_{\tau\tau} | \mu, \hat{\theta})}{L(\vec{N}_{ZZ}, \vec{N}_{WW}, \vec{N}_{\tau\tau} | \hat{\mu}, \hat{\theta})}$$

$$p(H_{\mu}) = \int_{\lambda_{obs}}^{\infty} f(\lambda | H_{\mu}) d\lambda = \dots$$

Model Usage phase (use $L(x|H)$ to make statement on H)

How is Higgs discovery different from a simple fit?

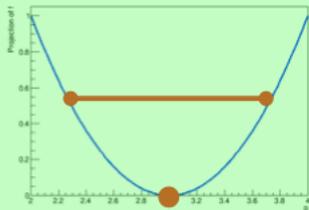
Gaussian + polynomial

Higgs combination model

Design goal:

Separate **building of Likelihood model** as much as possible from statistical analysis **using the Likelihood model**

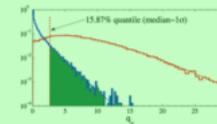
- More modular software design
- 'Plug-and-play with statistical techniques
- Factorizes work in collaborative effort



ML estimation of parameters μ, θ using MINUIT (MIGRAD, HESSE, MINOS)

$$\mu = 5.3 \pm 1.7$$

$$\lambda_{\mu}(\tilde{N}_{ZZ}, \tilde{N}_{WW}, \tilde{N}_{\tau\tau}) = \frac{L(\tilde{N}_{ZZ}, \tilde{N}_{WW}, \tilde{N}_{\tau\tau} | \mu, \hat{\theta})}{L(\tilde{N}_{ZZ}, \tilde{N}_{WW}, \tilde{N}_{\tau\tau} | \hat{\mu}, \hat{\theta})}$$



$$p(H_{\mu}) = \int_{\lambda_{\text{obs}}}^{\infty} f(\lambda | H_{\mu}) d\lambda = \dots$$

The idea behind the design of RooFit/RooStats/HistFactory

- Modularity, Generality and flexibility
- Step 1 – Construct the likelihood function $L(x|p)$

RooFit, or RooFit+HistFactory

- Step 2 – Statistical tests on parameter of interest p

Procedure can be Bayesian, Frequentist, or Hybrid),
but always based on $L(x|p)$

RooStats

- Steps 1 and 2 are conceptually separated,
and in Roo* suit also implemented separately.

The idea behind the design of RooFit/RooStats/HistFactory

- Steps 1 and 2 can be 'physically' separated (in time, or user)
- **Step 1** – Construct the likelihood function $L(x|p)$

RooFit, or RooFit+HistFactory



- **Step 2** – Statistical tests on parameter of interest p

RooStats

The benefits of modularity

- Perform different statistical test on exactly the same model

RooFit, or RooFit+HistFactory



RooWorkspace



“Simple fit”

(ML Fit with
HESSE or
MINOS)



RooStats

(Frequentist
with toys)



RooStats

(Frequentist
asymptotic)



RooStats

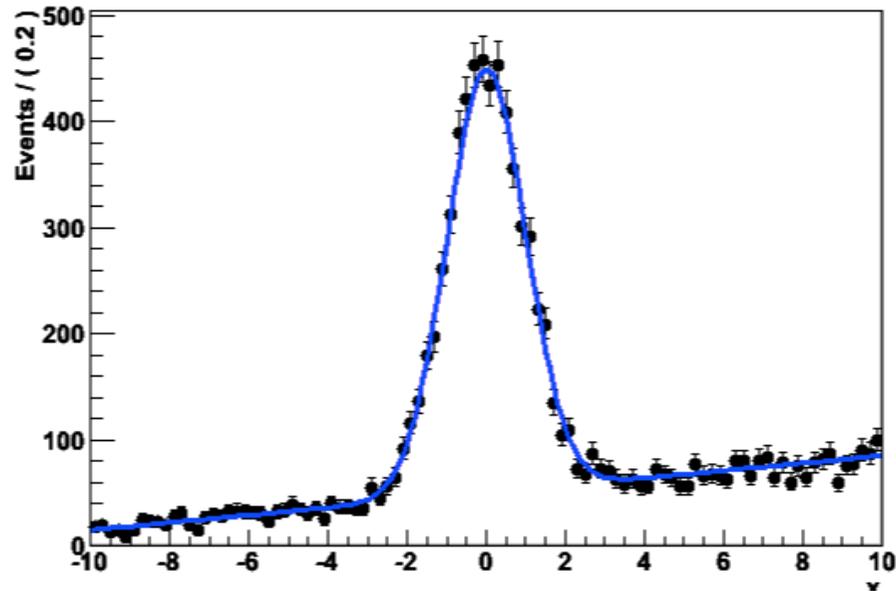
Bayesian
MCMC

RooFit

WV + D. Kirkby - 1999

RooFit – Focus: coding a probability density function

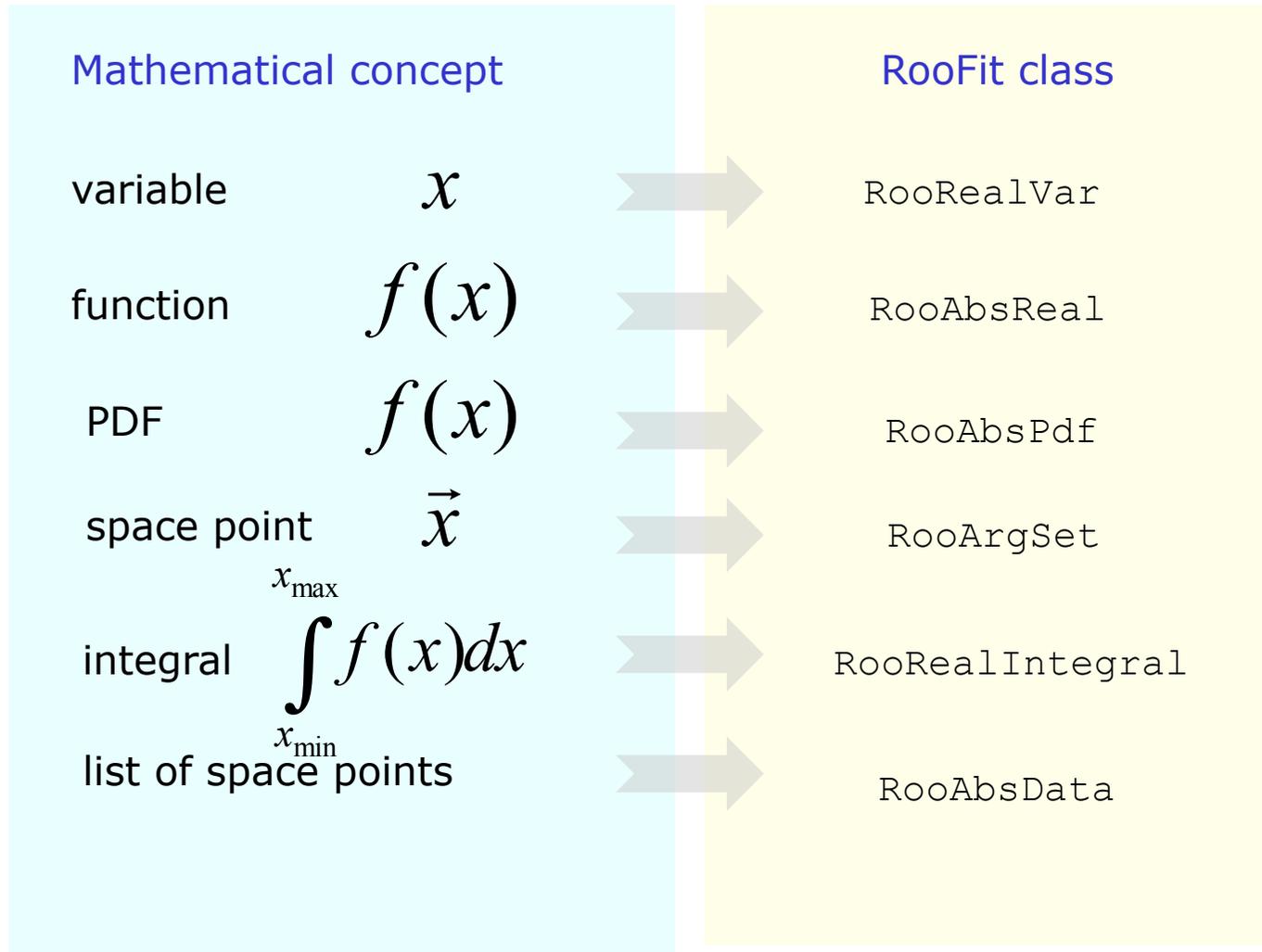
- Focus on one practical aspect of many data analysis in HEP:
How do you formulate your p.d.f. in ROOT
 - For ‘simple’ problems (gauss, polynomial) this is easy
 - But if you want to do unbinned ML fits, use non-trivial functions, or work with multidimensional functions you quickly find that you need some tools to help you



- The RooFit project started in 1999 for data modeling needs for BaBar collaboration initially, publicly available in ROOT since 2003

RooFit core design philosophy

- Mathematical objects are represented as C++ objects



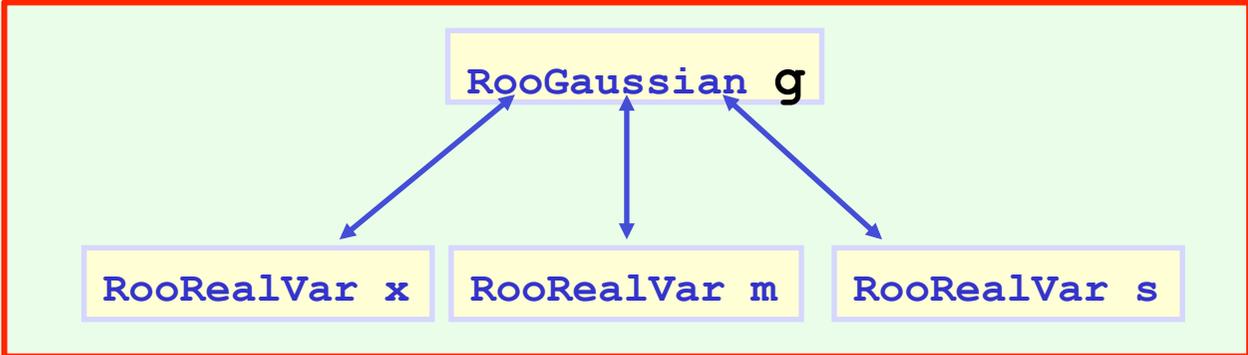
Data modeling – Constructing composite objects

- Straightforward correlation between mathematical representation of formula and RooFit code

Math	$gauss(x, m, \sqrt{s})$
RooFit diagram	<pre>graph TD x["RooRealVar x (1)"] --> g["RooGaussian g (5)"] m["RooRealVar m (2)"] --> g sqrt_s["RooFormulaVar sqrt(s) (4)"] --> g s["RooRealVar s (3)"] --> sqrt_s</pre>
RooFit code	<pre>① RooRealVar x("x", "x", -10, 10) ; ② RooRealVar m("m", "mean", 0) ; ③ RooRealVar s("s", "sigma", 2, 0, 10) ; ④ RooFormulaVar sqrt_s("sqrt(s)", "sqrt(s)", s) ; ⑤ RooGaussian g("g", "gauss", x, m, sqrt_s) ;</pre>

RooFit core design philosophy

- A special container class owns all objects that together build a likelihood function

Math	$\text{Gauss}(x, \mu, \sigma)$
RooFit diagram	<p>RooWorkspace (keeps all parts together)</p>  <pre> graph TD subgraph RooWorkspace x[RooRealVar x] m[RooRealVar m] s[RooRealVar s] g[RooGaussian g] x --> g m --> g s --> g g <--> m end </pre>
RooFit code	<pre> RooRealVar x("x","x",-10,10) ; RooRealVar m("m","y",0,-10,10) ; RooRealVar s("s","z",3,0.1,10) ; RooGaussian g("g","g",x,m,s) ; RooWorkspace w("w") ; w.import(g) ; </pre>

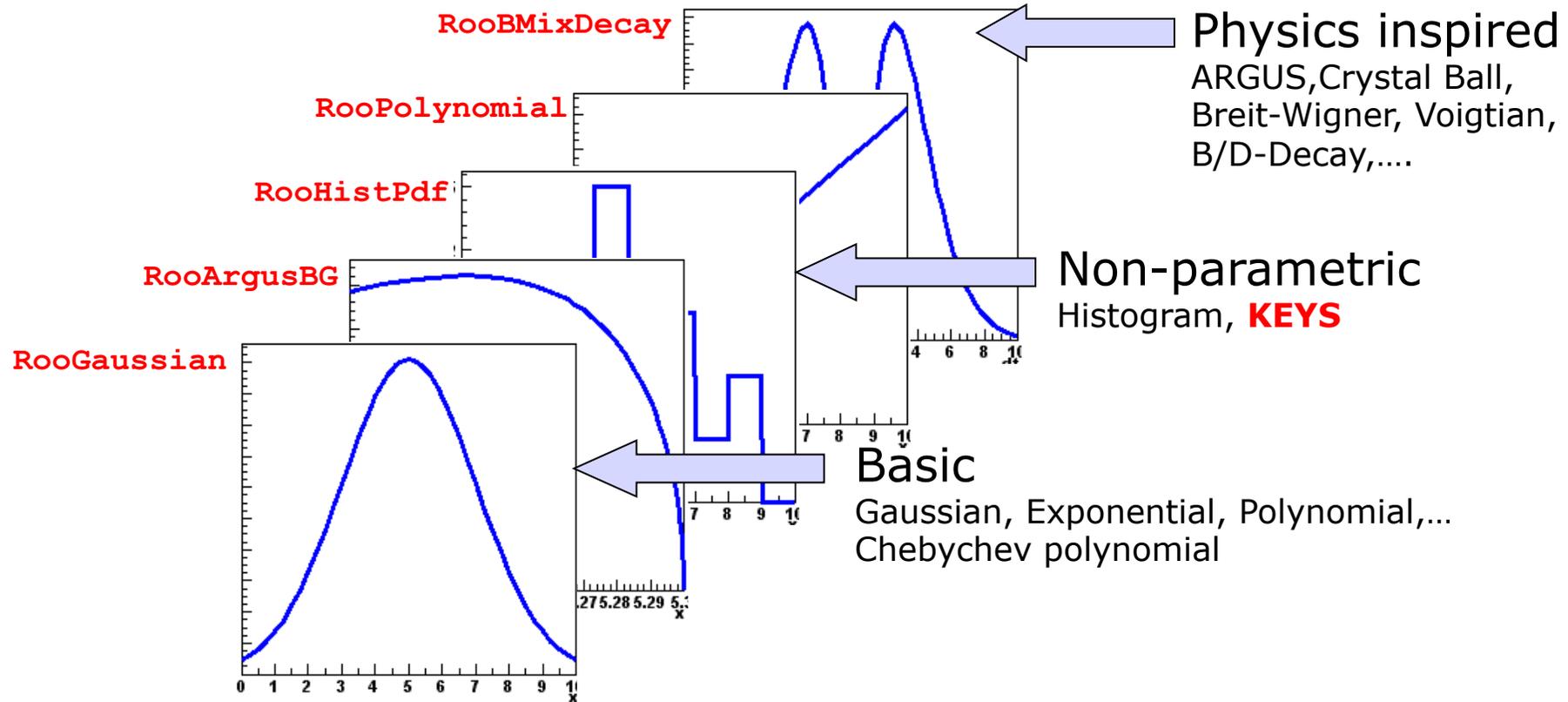
Populating a workspace the easy way – “the factory”

- The **factory** allows to fill a workspace with pdfs and variables using a simplified scripting language

Math	$\text{Gauss}(x, \mu, \sigma)$ <i>New feature for LHC</i>
RooFit diagram	<pre>graph BT; x[RooRealVar x] --> f[RooAbsReal f]; y[RooRealVar y] <--> f; z[RooRealVar z] --> f;</pre>
RooFit code	<pre>RooWorkspace w("w") ; w.factory("Gaussian::g(x[-10,10],m[-10,10],z[3,0.1,10])") ;</pre>

Model building – (Re)using standard components

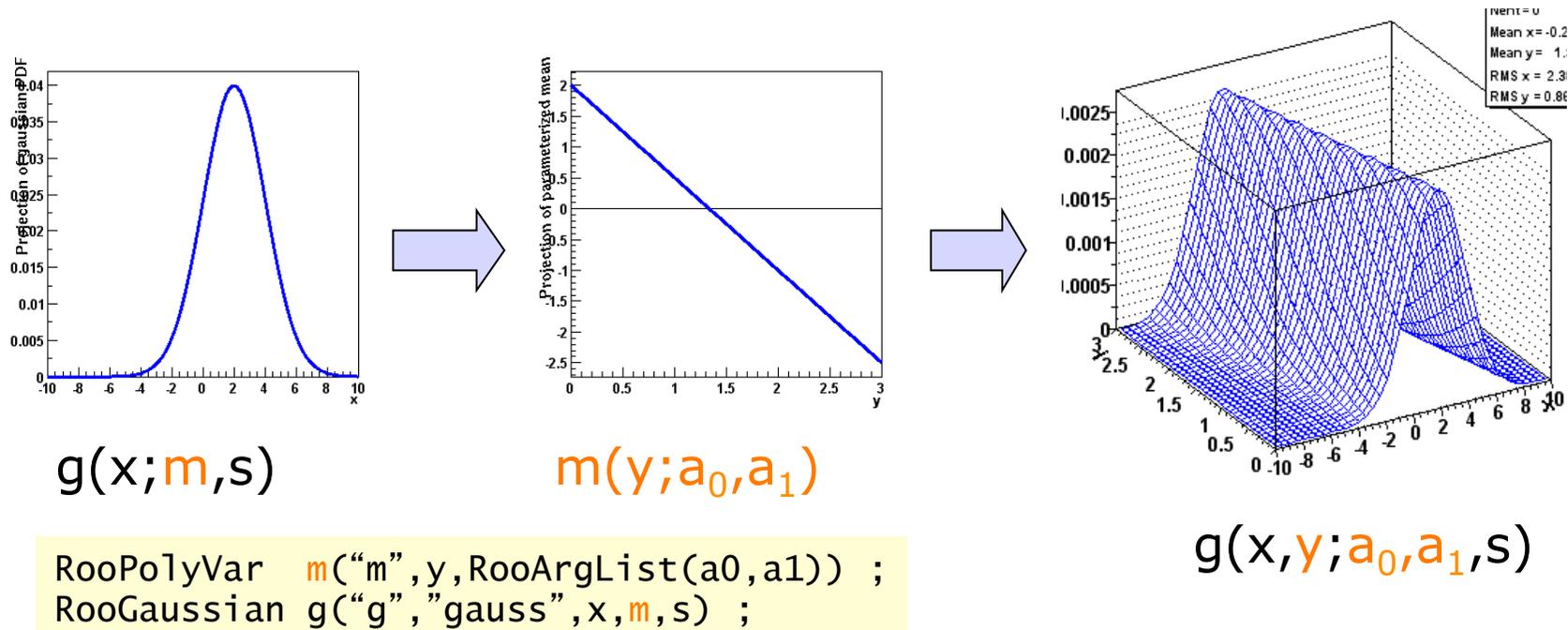
- RooFit provides a collection of compiled standard PDF classes



Easy to extend the library: each p.d.f. is a separate C++ class

Model building – (Re)using standard components

- Library p.d.f.s can be adjusted on the fly.
 - Just plug in *any function expression* you like as input variable
 - Works universally, even for classes you write yourself

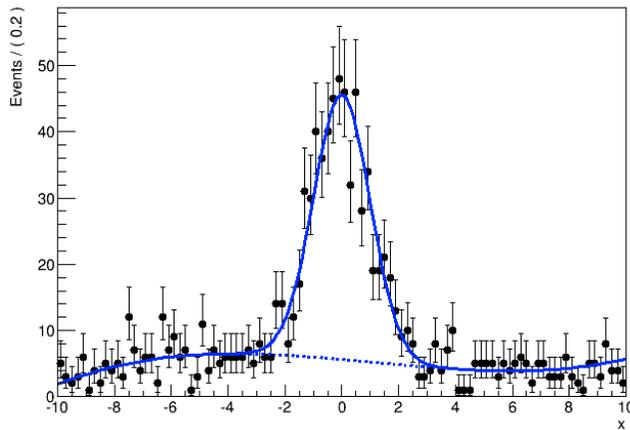


- Maximum flexibility of library shapes keeps library small

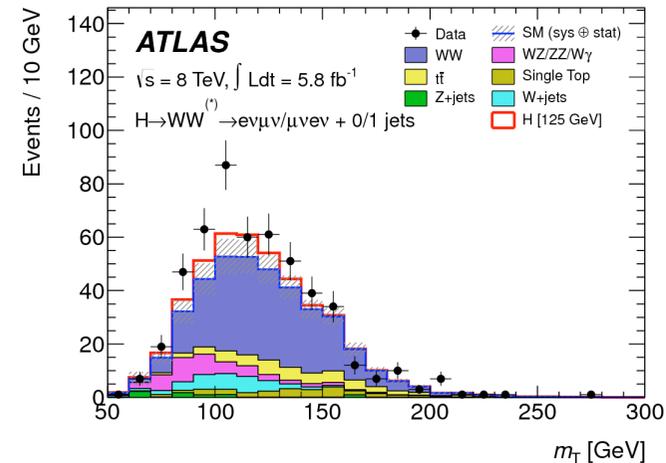
From empirical probability models to simulation-based models

- Large difference between B-physics and LHC hadron physics is that for the latter distributions usually don't follow simple analytical shapes

Unbinned analytical probability model

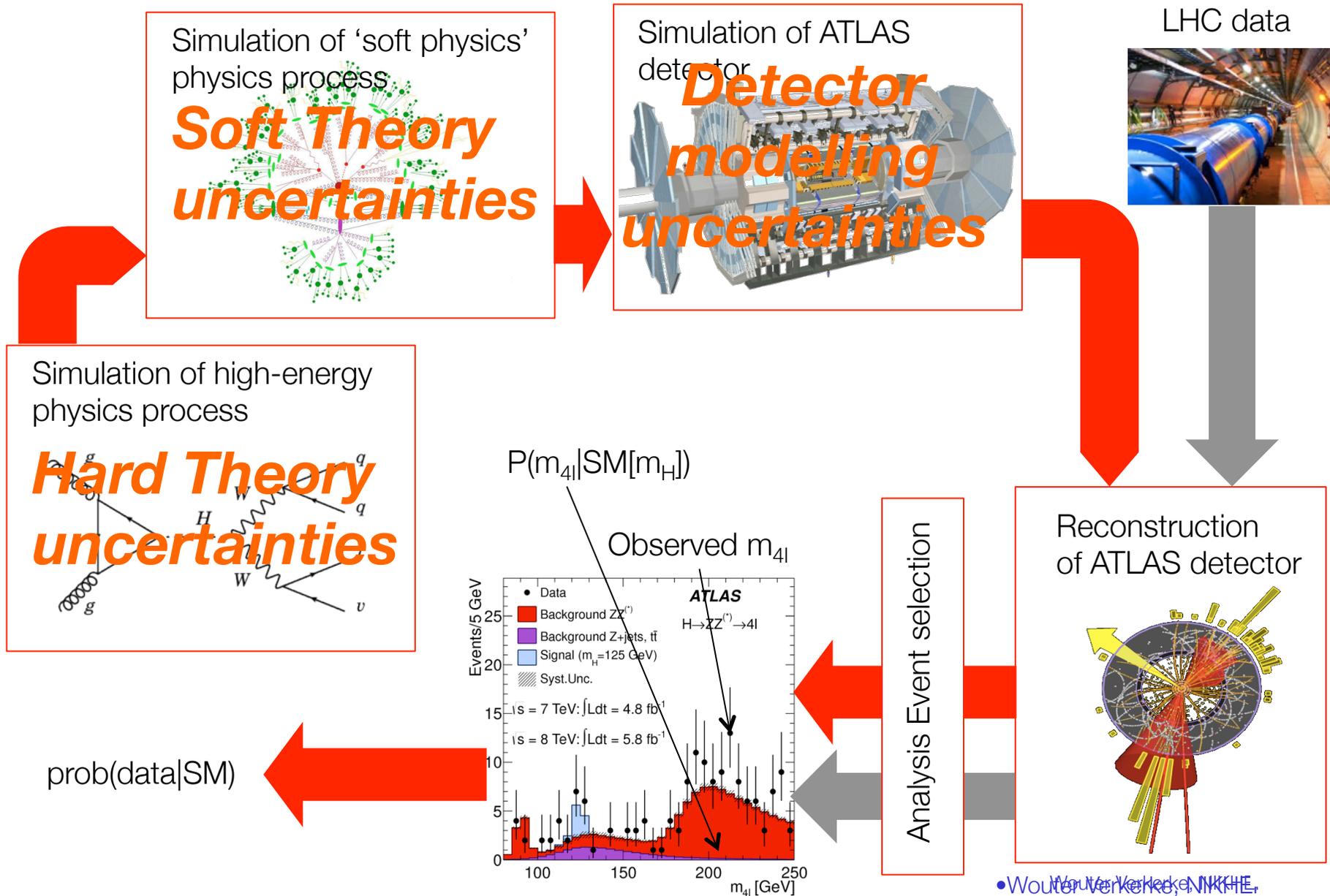


(Geant) Simulation-driven binned template model



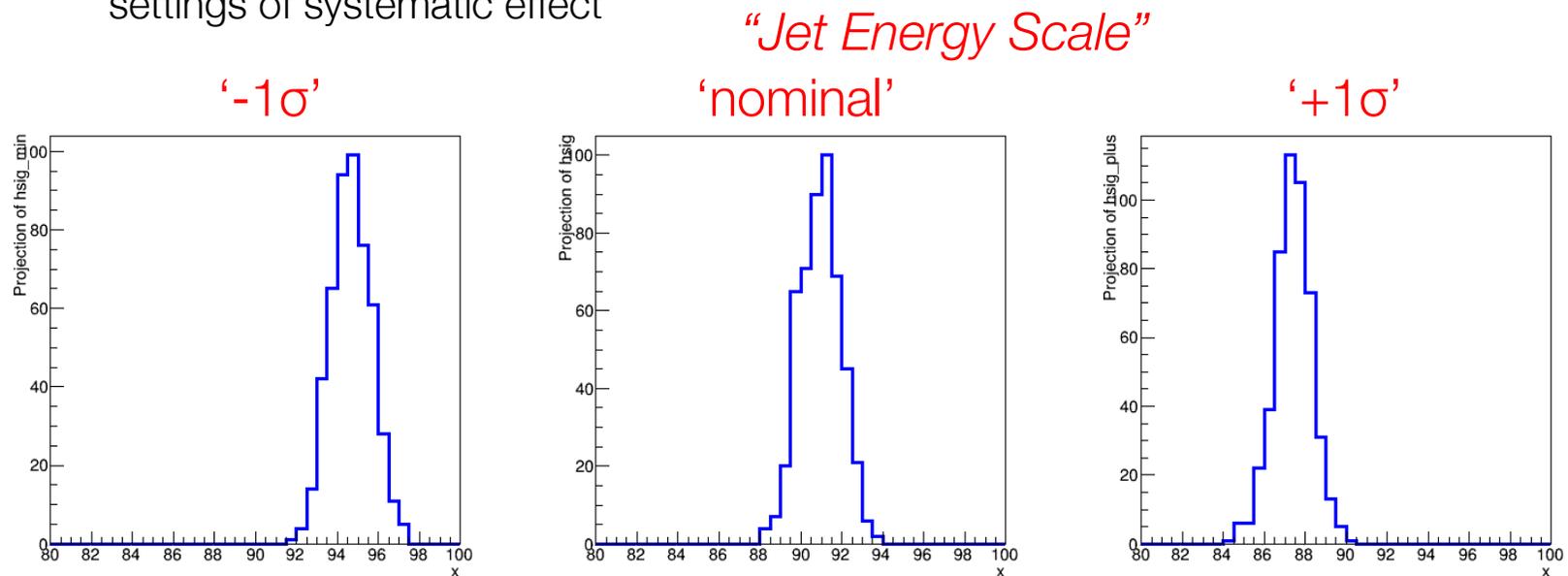
- But concept of simulation-driven template models can also extend to systematic uncertainties. Instead of empirically chosen 'nuisance parameters' (e.g. polynomial coefs) construct degrees of freedom that correspond to known systematic uncertainties

The HEP analysis workflow illustrated



Modeling of shape systematics in the likelihood

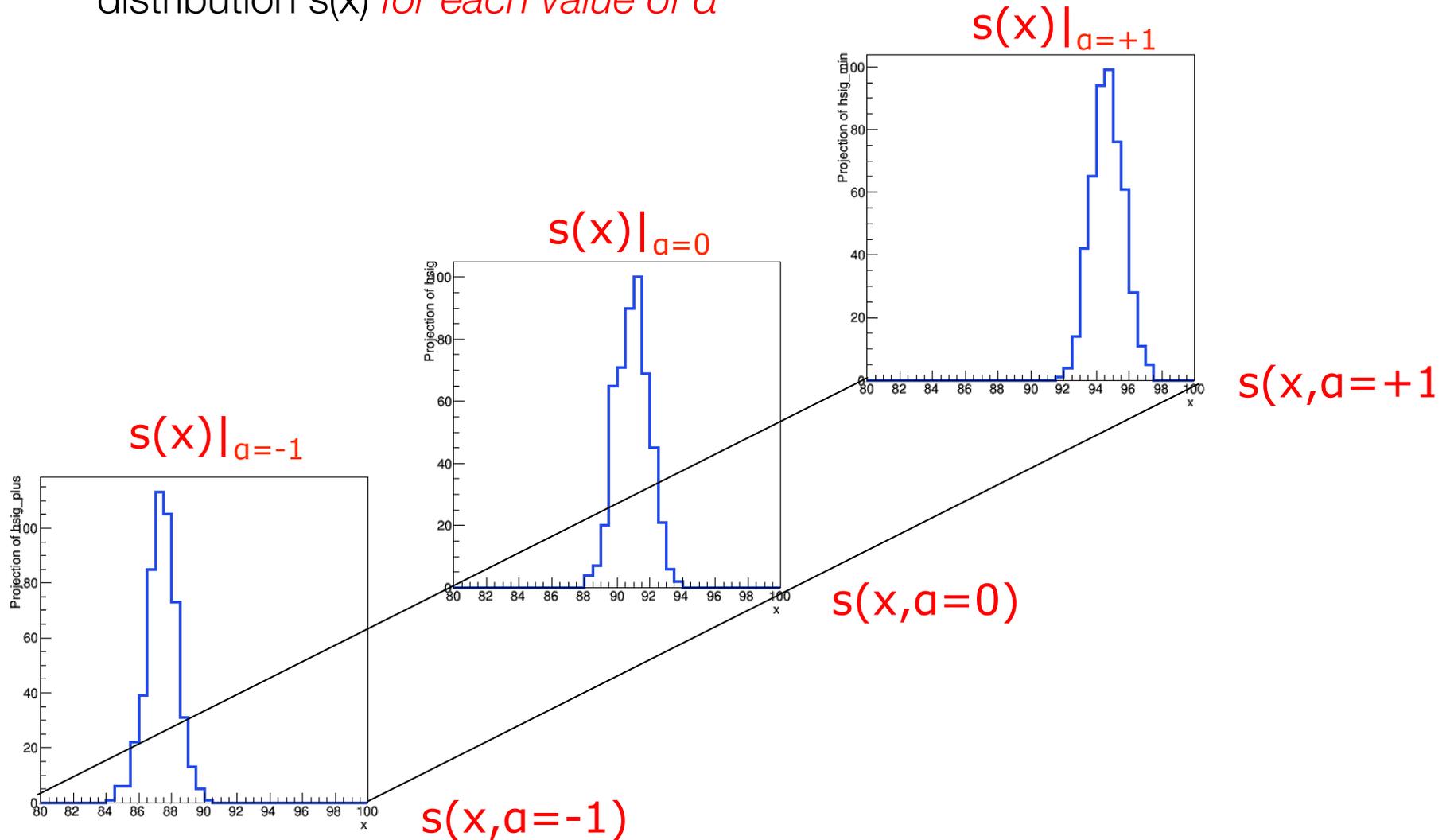
- Effect of *any* systematic uncertainty that affects the shape of a distribution can in principle be obtained from MC simulation chain
 - Obtain histogram templates for distributions at ‘+1 σ ’ and ‘-1 σ ’ settings of systematic effect



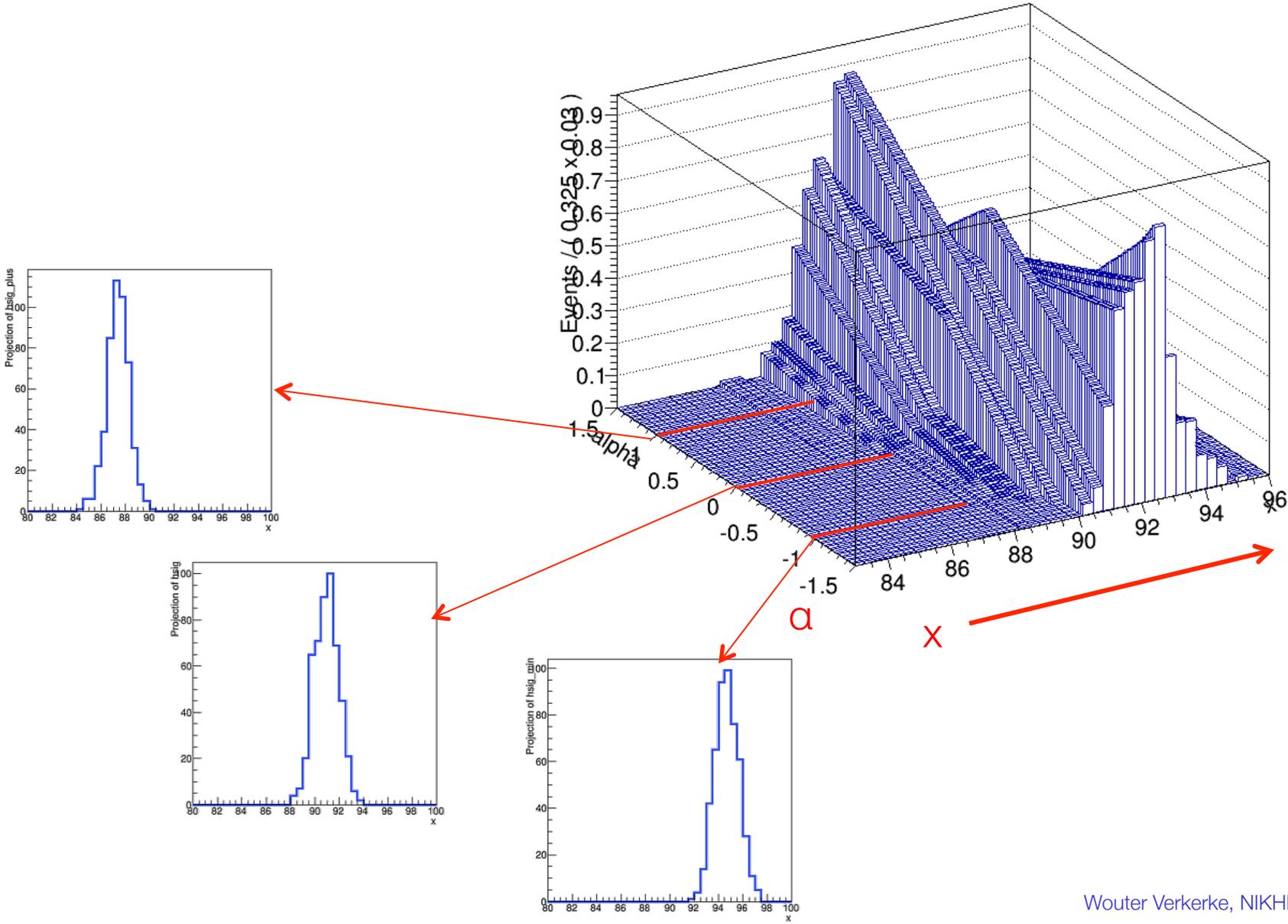
- Challenge: **construct an empirical response function based on the interpolation of the shapes of these three templates.**

Need to interpolate between template models

- Need to define ‘morphing’ algorithm to define distribution $s(x)$ *for each value of a*



Visualization of bin-by-bin linear interpolation of distribution

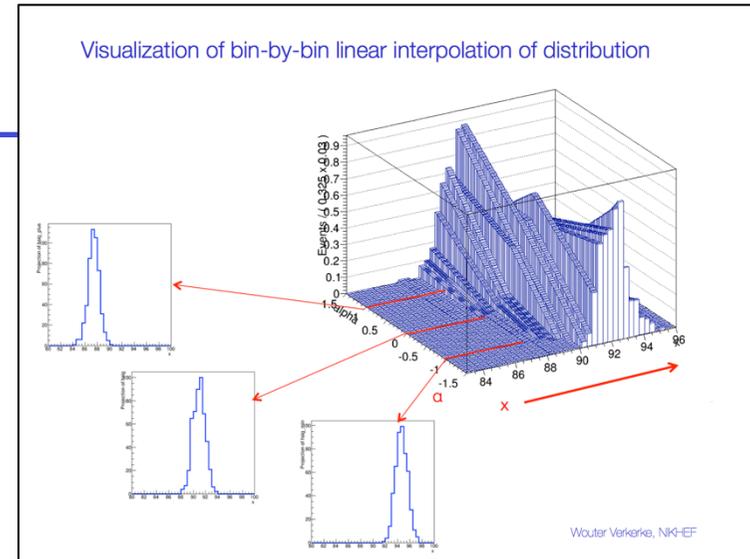


Example 2 : binned L with syst

- Example of template morphing systematic in a binned likelihood

$$s_i(\alpha, \dots) = \begin{cases} s_i^0 + \alpha \cdot (s_i^+ - s_i^0) & \forall \alpha > 0 \\ s_i^0 + \alpha \cdot (s_i^0 - s_i^-) & \forall \alpha < 0 \end{cases}$$

$$L(\vec{N} | \alpha, \vec{s}^-, \vec{s}^0, \vec{s}^+) = \prod_{bins} P(N_i | \underbrace{s_i(\alpha, s_i^-, s_i^0, s_i^+)}_{\text{red}}) \cdot \underbrace{G(0 | \alpha, 1)}_{\text{green}}$$



```
// Import template histograms in workspace
w.import(hs_0,hs_p,hs_m) ;

// Construct template models from histograms
w.factory("HistFunc::s_0(x[80,100],hs_0)" ) ;
w.factory("HistFunc::s_p(x,hs_p)" ) ;
w.factory("HistFunc::s_m(x,hs_m)" ) ;

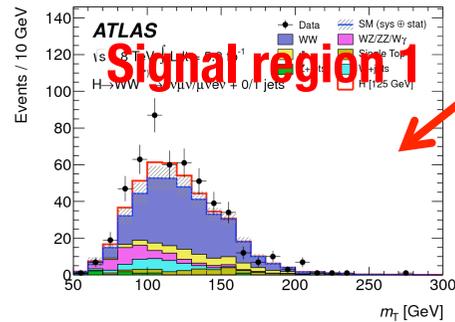
// Construct morphing model
w.factory("PiecewiseInterpolation::sig(s_0,s_m,s_p,alpha[-5,5])" ) ;

// Construct full model
w.factory("PROD::model(ASUM(sig,bkg,f[0,1]),Gaussian(0,alpha,1))" ) ;
```

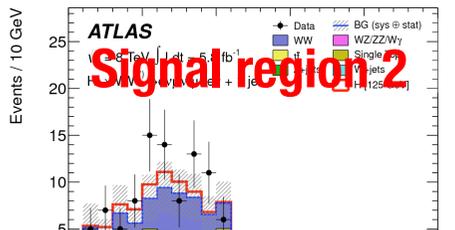
The structure of an (Higgs) profile likelihood function

- Likelihood describing Higgs samples have following structure

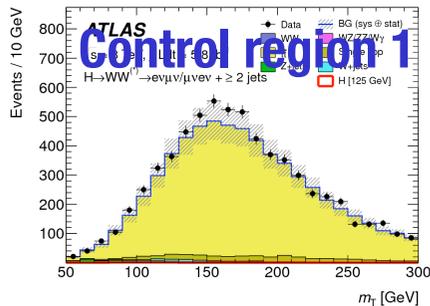
$$L_{H \rightarrow X}(x | \mu, \vec{\theta}) = \prod_{i=0 \dots n} L_{phys}(x | \mu, \vec{\theta}) \cdot \prod_{i=0 \dots n} L_{control}(x | \mu, \vec{\theta}) \cdot L_{sub}(\theta_1) \cdot L_{sub}(\theta_1) \cdots L_{sub}(\theta_n)$$



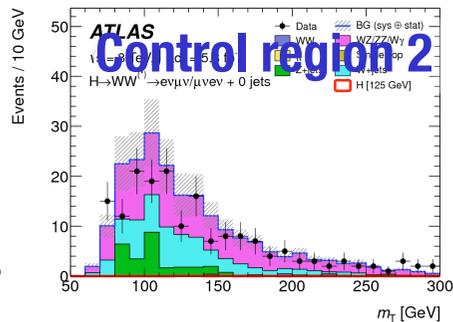
Signal region 1



Signal region 2



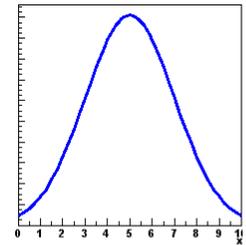
Control region 1



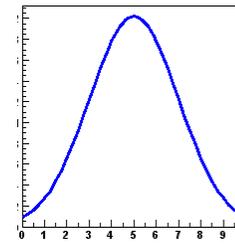
Control region 2

Strength of systematic uncertainties

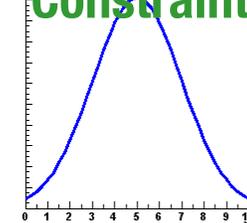
'Constraint θ_n '



'Constraint θ_1 '



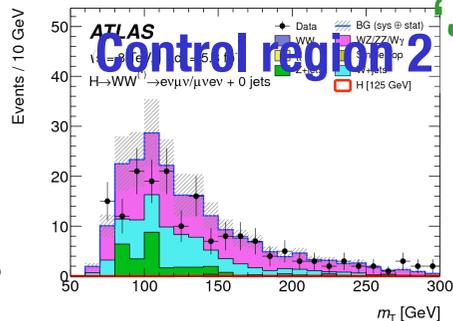
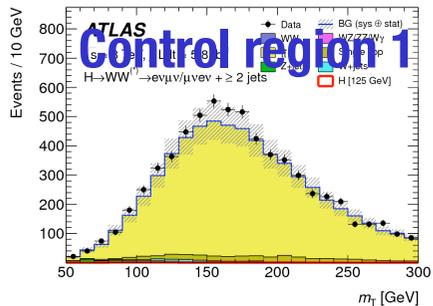
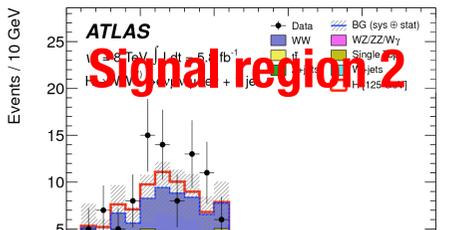
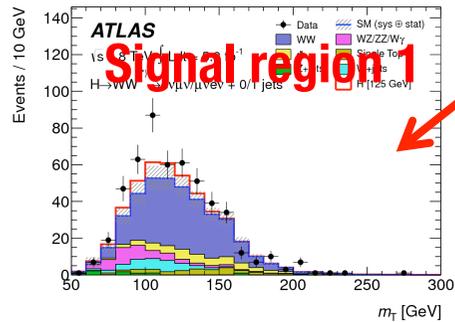
'Constraint θ_n '



The structure of an (Higgs) profile likelihood function

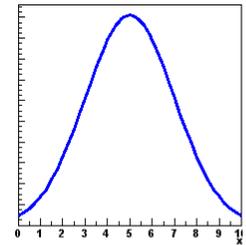
- A simultaneous fit of physics samples and (simplified) performance measurements

$$L_{H \rightarrow X}(x | \mu, \vec{\theta}) = \prod_{i=0 \dots n} L_{phys}(x | \mu, \vec{\theta}) \cdot \prod_{i=0 \dots n} L_{control}(x | \mu, \vec{\theta}) \cdot L_{sub}(\theta_1) \cdot L_{sub}(\theta_1) \cdot \dots \cdot L_{sub}(\theta_n)$$



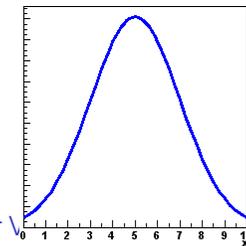
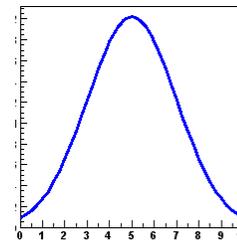
'Simplified Likelihood of a measurement related to systematic uncertainties'

'Subsidiary measurement n'
Factorization scale



'Subsidiary measurement 1' **'Subsidiary measurement 2'**

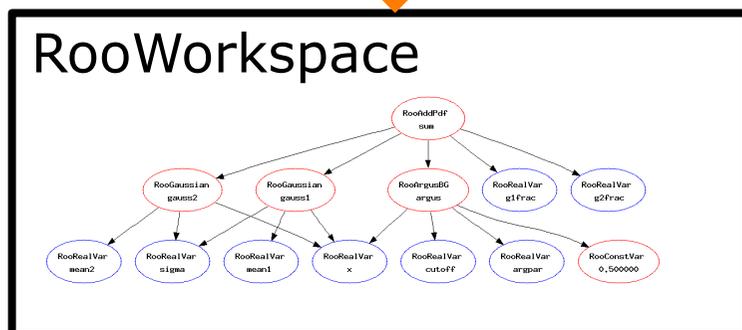
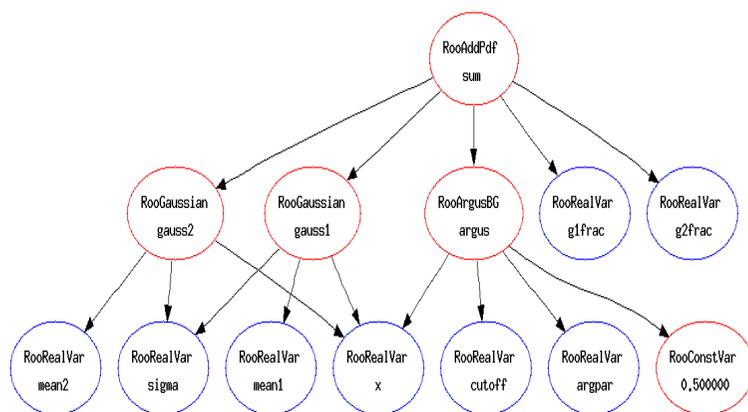
'Jet Energy scale' **B-tagging eff**



The Workspace

The workspace

- The workspace concept has revolutionized the way people share and combine analysis
 - **Completely** factorizes process of building and using likelihood functions
 - You can give somebody an analytical likelihood of a (potentially very complex) physics analysis in a way to the easy-to-use, provides introspection, and is easy to modify.

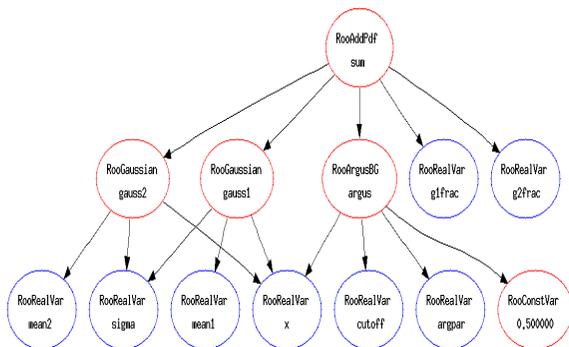
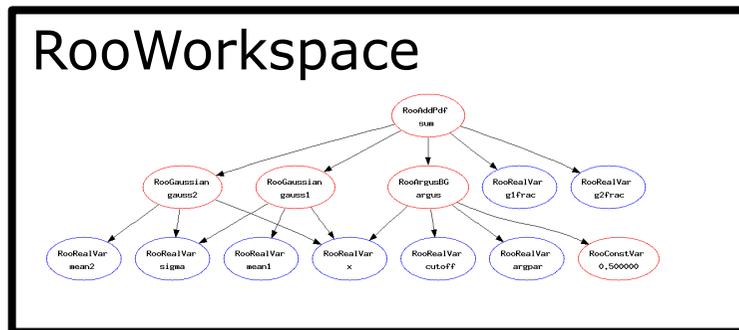
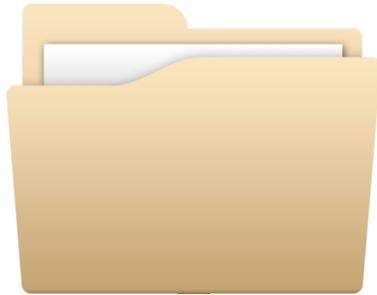


```
RooWorkspace w("w") ;  
w.import(sum) ;  
w.writeToFile("model.root") ;
```

model.root

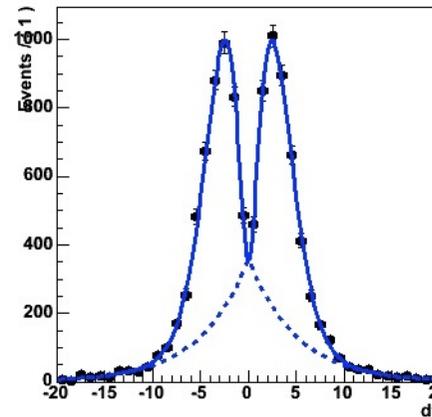


Using a workspace



```
// Resurrect model and data
TFile f("model.root") ;
RooWorkspace* w = f.Get("w") ;
RooAbsPdf* model = w->pdf("sum") ;
RooAbsData* data = w->data("xxx") ;
```

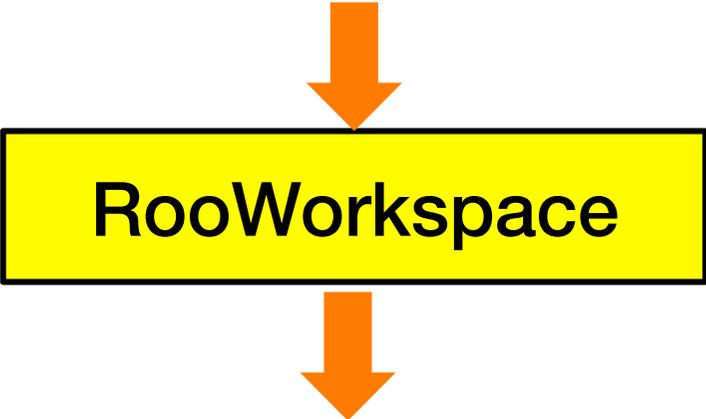
```
// Use model and data
model->fitTo(*data) ;
RooPlot* frame =
    w->var("dt")->frame() ;
data->plotOn(frame) ;
model->plotOn(frame) ;
```



The idea behind the design of RooFit/RooStats/HistFactory

- **Step 1** – Construct the likelihood function $L(x|p)$

```
RooWorkspace w("w") ;  
w.factory("Gaussian::sig(x[-10,10],m[0],s[1])") ;  
w.factory("Chebychev::bkg(x,a1[-1,1])") ;  
w.factory("SUM::model(fsig[0,1]*sig,bkg)") ;  
w.writeToFile("L.root") ;
```



RooWorkspace

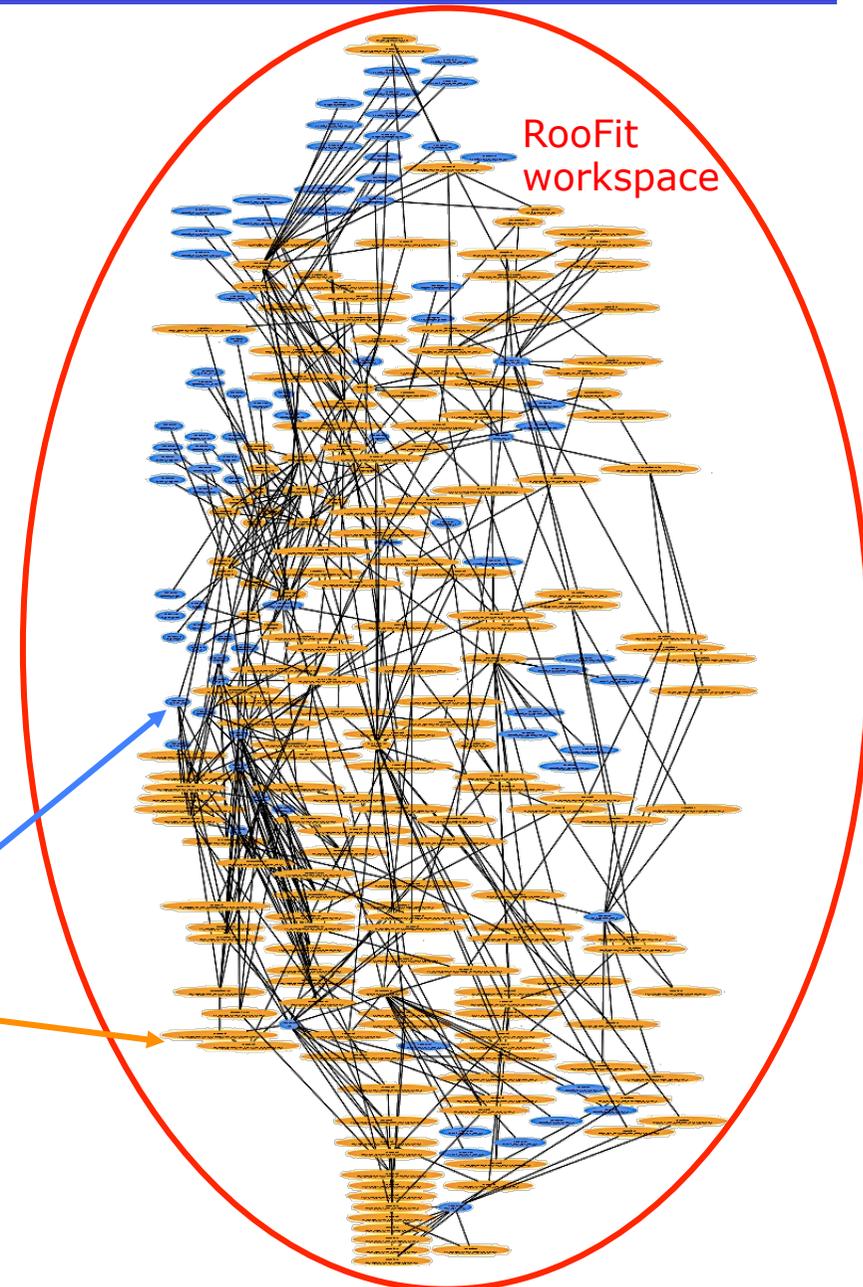
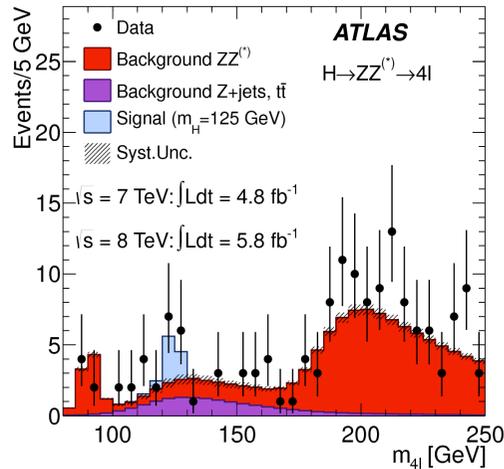
*Complete description
of likelihood model,
persistable in ROOT file
(RooFit pdf function)
Allows full introspection
and a-posteriori editing*

- **Step 2** – Statistical tests on parameter of interest p

```
RooWorkspace* w=TFile::Open("L.root")->Get("w") ;  
RooAbsPdf* model = w->pdf("model") ;  
pdf->fitTo(data) ;
```

Example RooFit component model for realistic Higgs analysis

Likelihood model describing the ZZ invariant mass distribution including all possible systematic uncertainties



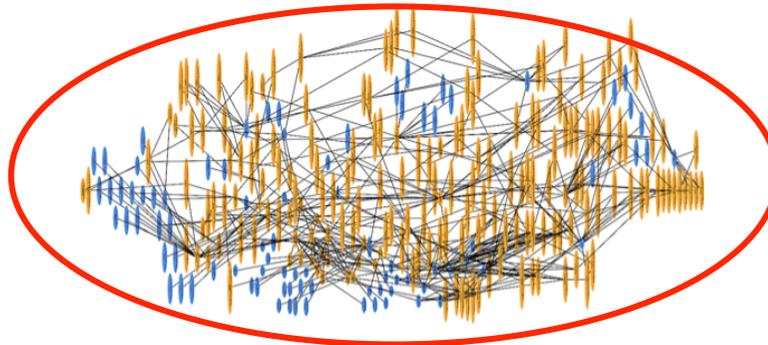
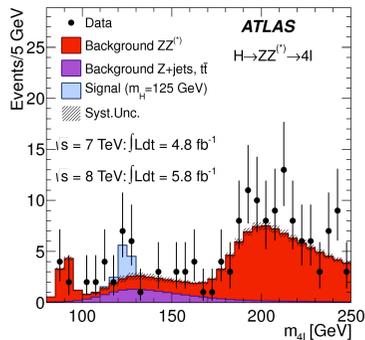
variables

function objects

Graphical illustration of function components that call each other

Analysis chain identical for highly complex (Higgs) models

- **Step 1** – Construct the likelihood function $L(x|p)$



Rooworkspace

*Complete description
of likelihood model,
persistable in ROOT file
(RooFit pdf function)
Allows full introspection
and a-posteriori editing*

- **Step 2** – Statistical tests on parameter of interest p

```
Rooworkspace* w=TFile::Open("L.root")->Get("w") ;  
RooAbsPdf* model = w->pdf("model") ;  
pdf->fitTo(data,  
            GlobalObservables(w->set("MC_Globs"),  
            Constrain(*w->st("MC_NuisParams")) ;
```

Workspaces power collaborative statistical modelling

- Ability to persist complete^(*) Likelihood models has profound implications for HEP analysis workflow
 - (*) Describing signal regions, control regions, and including nuisance parameters for all systematic uncertainties)
- **Anyone with ROOT (and one ROOT file with a workspace) can re-run any entire statistical analysis out-of-the-box**
 - About 5 lines of code are needed
 - Including estimate of systematic uncertainties
- Unprecedented new possibilities for cross-checking results, in-depth checks of structure of analysis
 - Trivial to run variants of analysis (what if 'Jet Energy Scale uncertainty' is 7% instead of 4%). Just change number and rerun.
 - But can also make structural changes a posteriori. For example, rerun with assumption that JES uncertainty in forward and barrel region of detector are 100% correlated instead of being uncorrelated.

Collaborative statistical modelling

- As an experiment, you can effectively **build a library of measurements**, of which the full likelihood model is preserved for later use
 - Already done now, experiments have such libraries of workspace files,
 - Archived in AFS directories, or even in SVN....
 - Version control of SVN, or numbering scheme in directories allows for easy validation and debugging as new features are added
- Building of combined likelihood models greatly simplified.
 - Start from persisted components. No need to (re)build input components.
 - No need to know how individual components were built, or are internally structured. Just need to know meaning of parameters.
 - Combinations can be produced (much) later than original analyses.
 - Even analyses that were never originally intended to be combined with anything else can be included in joint likelihoods at a later time

Higgs discovery strategy – add everything together

$H \rightarrow ZZ \rightarrow 4l$

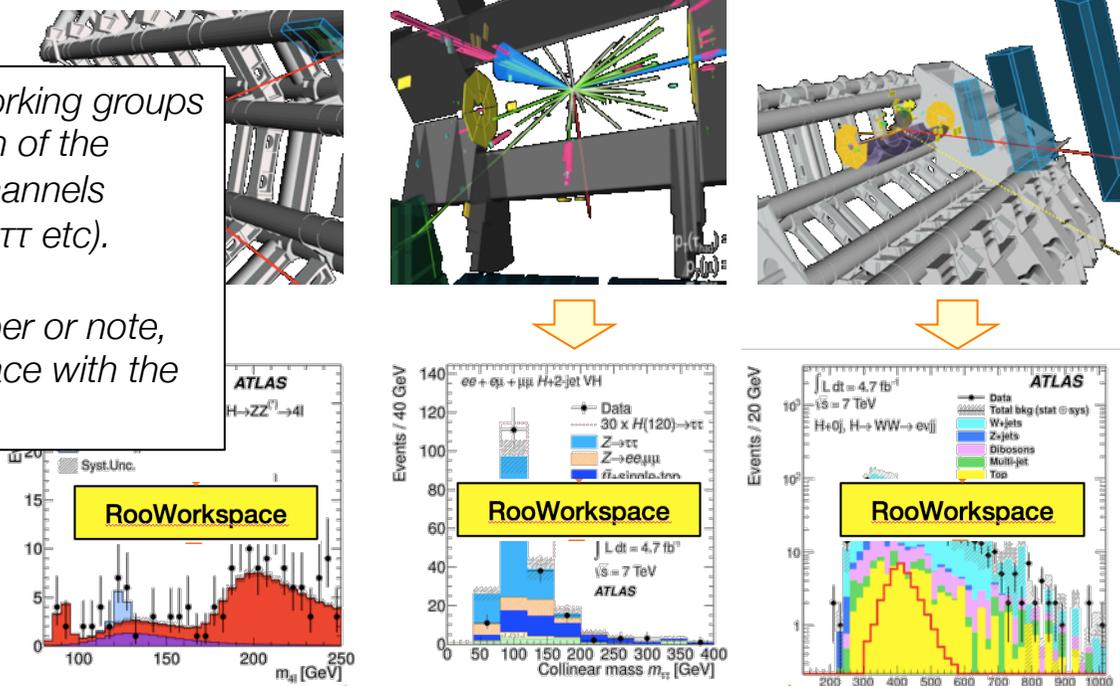
$H \rightarrow \tau\tau$

$H \rightarrow WW \rightarrow \mu\nu jj$

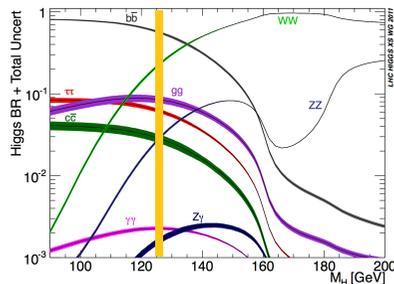
+ ...

Dedicated physics working groups define search for each of the major Higgs decay channels ($H \rightarrow WW$, $H \rightarrow ZZ$, $H \rightarrow \tau\tau$ etc).

Output is physics paper or note, and a RooFit workspace with the full likelihood function



Assume SM rates



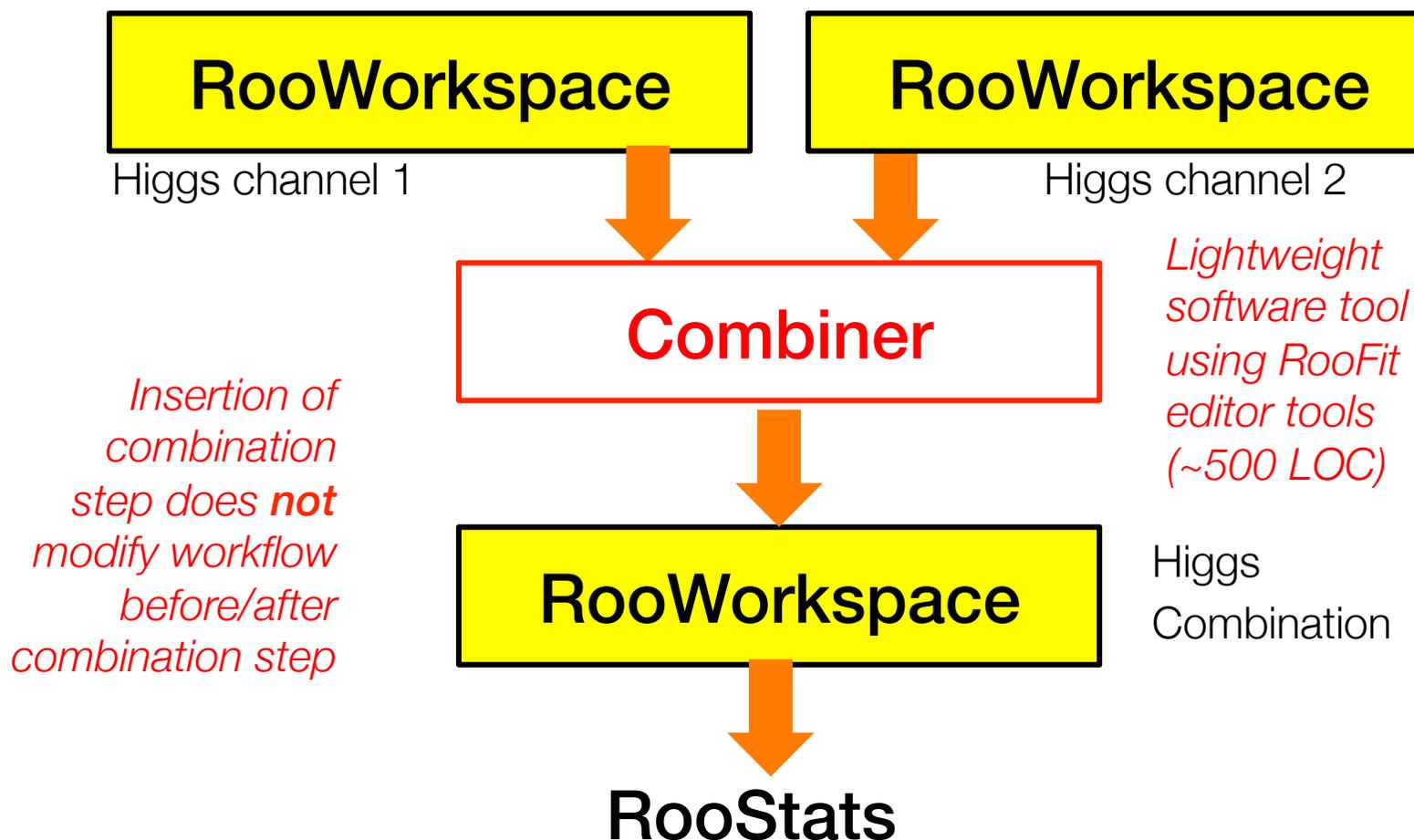
$$L(\mu, \vec{\theta}) = L_{H \rightarrow WW}(\mu_{WW}, \vec{\theta}) \cdot L_{H \rightarrow \gamma\gamma}(\mu_{\gamma\gamma}, \vec{\theta}) \cdot L_{H \rightarrow ZZ}(\mu_{ZZ}, \vec{\theta}) \cdot \dots$$

A small dedicated team of specialists builds a combined likelihood from the inputs. Major discussion point: naming of parameters, choice of parameters for systematic uncertainties (a physics issue, largely)

The benefits of modularity

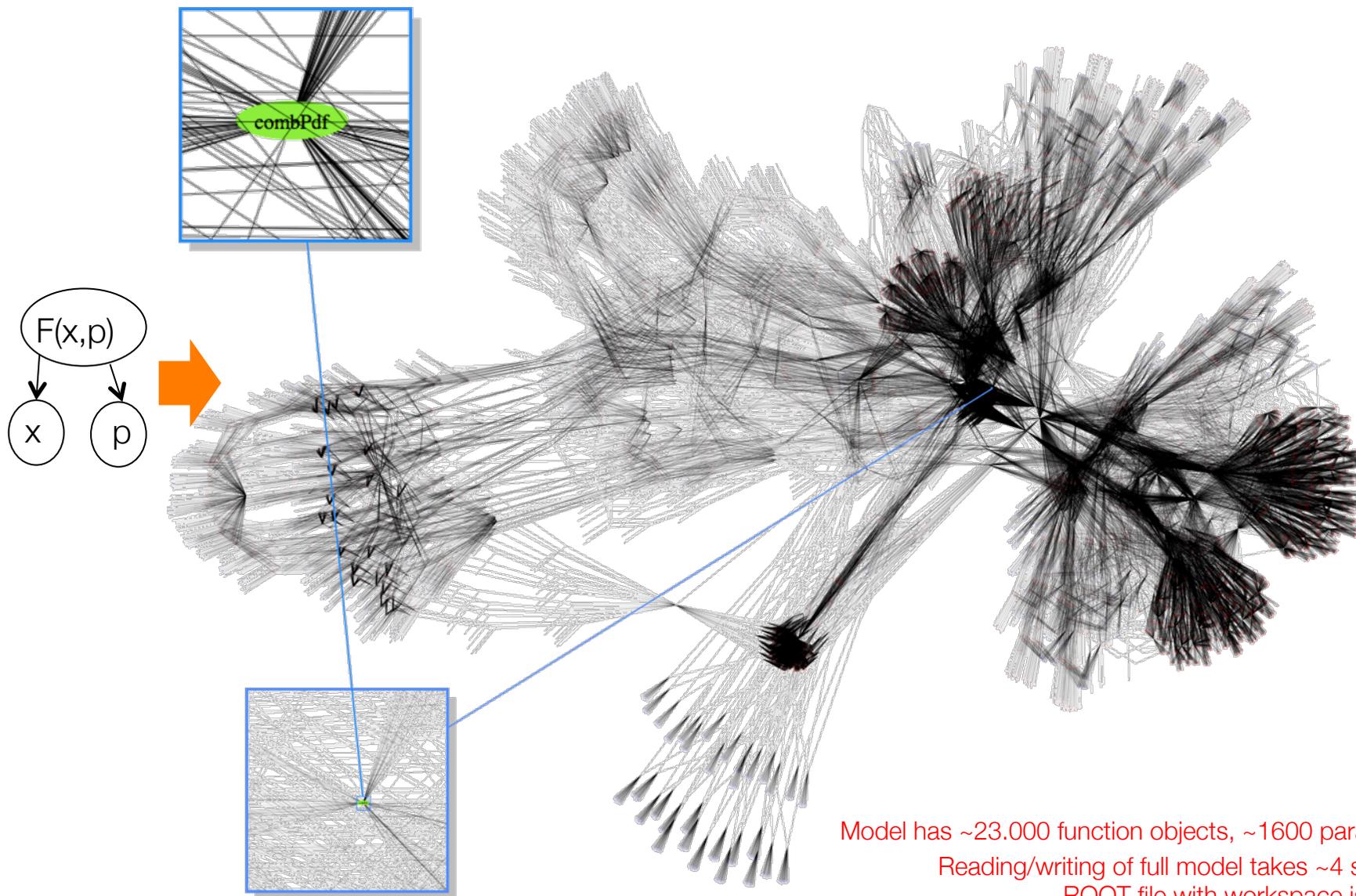
- Technically very straightforward to combine measurements

RooFit, or RooFit+HistFactory



Workspace persistence of *really* complex models works too!

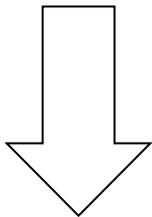
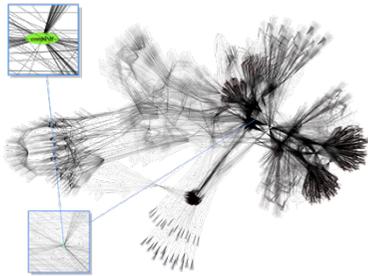
Atlas Higgs combination model (23.000 functions, 1600 parameters)



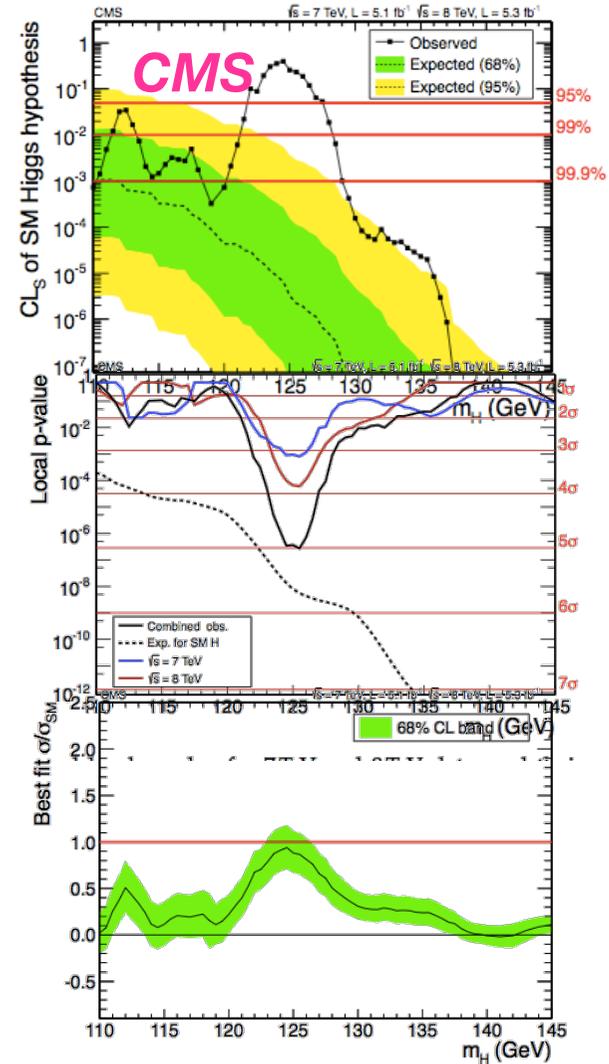
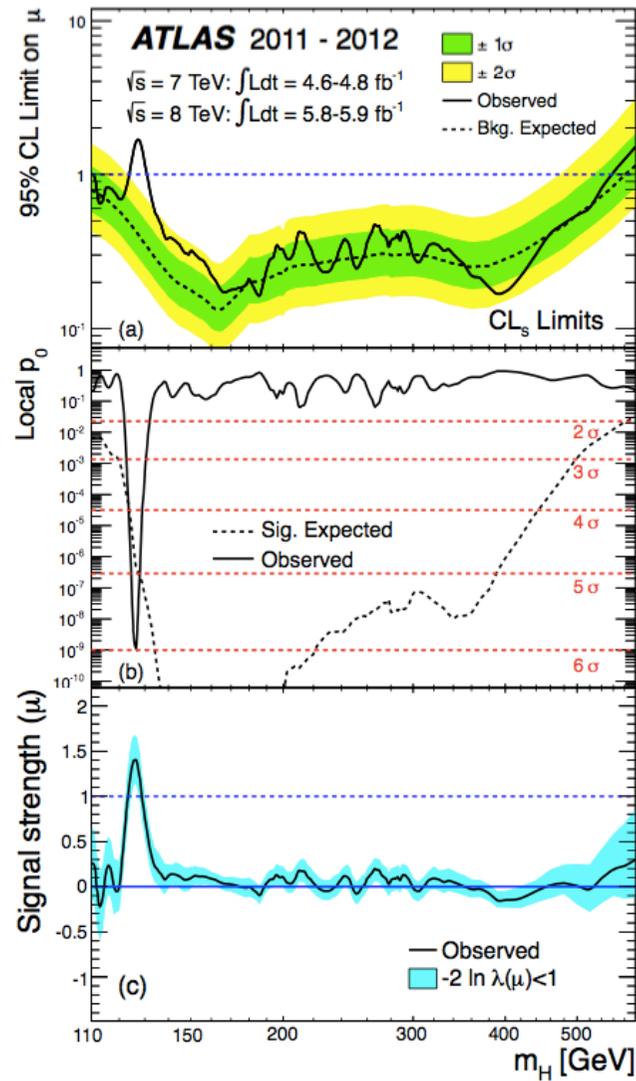
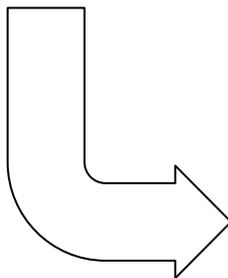
Model has ~23.000 function objects, ~1600 parameters
Reading/writing of full model takes ~4 seconds
ROOT file with workspace is ~6 Mb

With these combined models the Higgs discovery plots were produced...

$$L_{\text{ATLAS}}(\mu, \theta) =$$



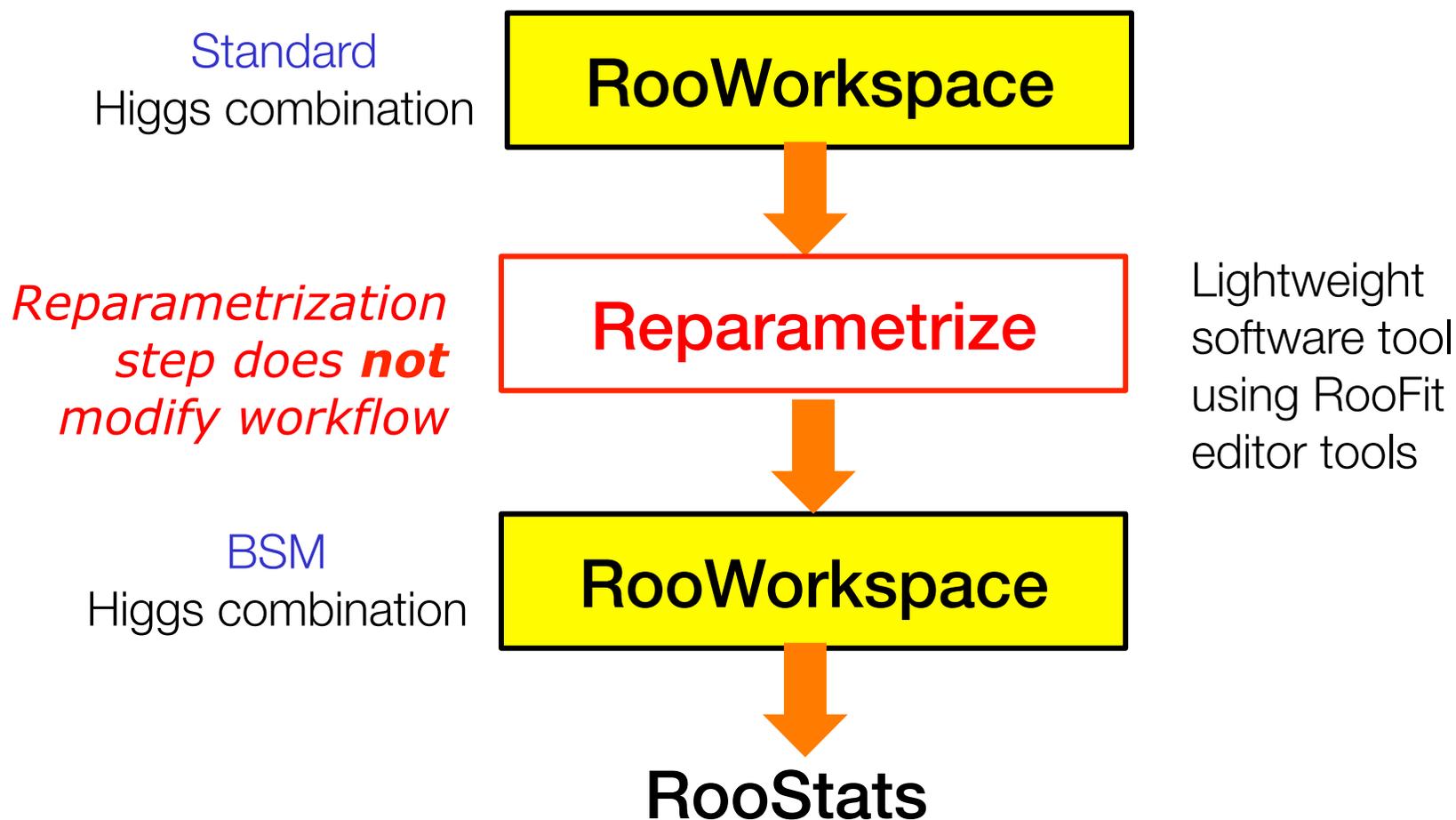
Neyman construction with profile likelihood ratio test



More benefits of modularity

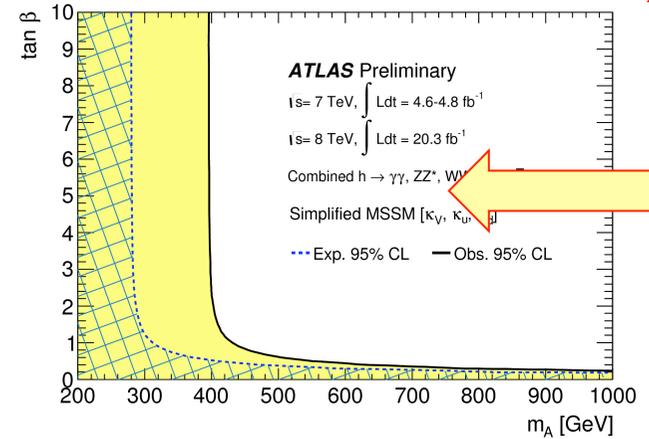
- Technically very straightforward to reparametrize measurements

RooFit, or RooFit+HistFactory

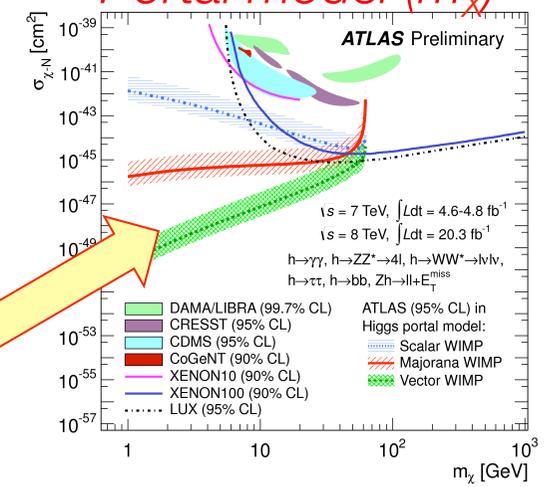


BSM Higgs constraints from reparametrization of SM Higgs Likelihood model

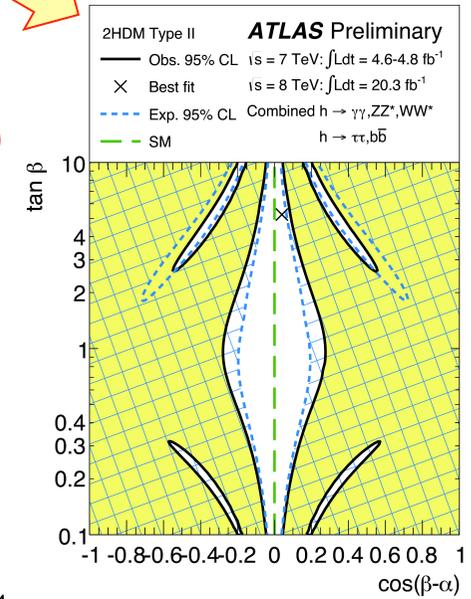
Simplified MSSM ($\tan\beta, m_A$)



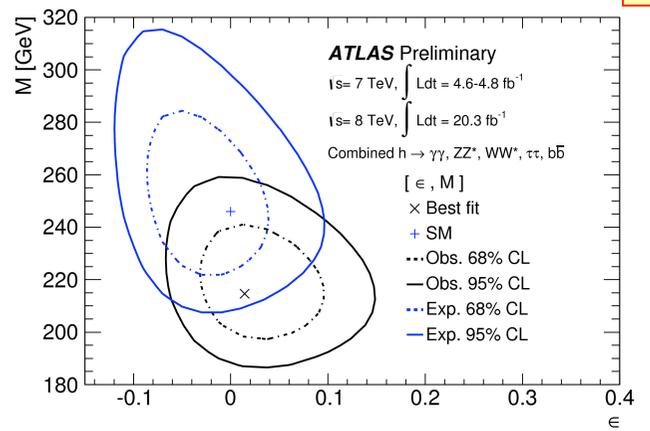
Portal model (m_χ)



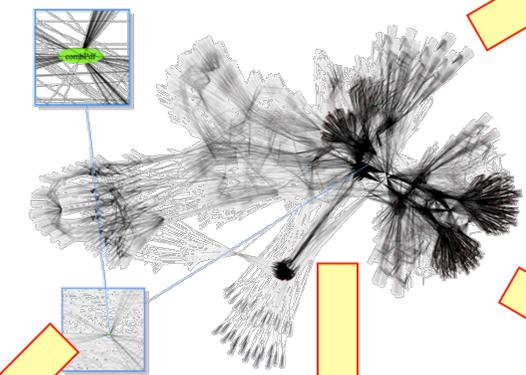
Two Higgs Double Model ($\tan\beta, \cos(\alpha-\beta)$)



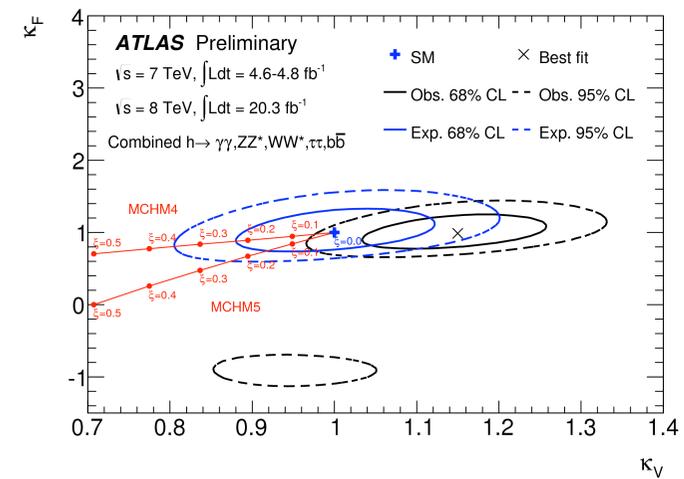
Imposter model (M, ϵ)



(ATLAS-CONF-2014-010)



Minimal composite Higgs (ξ)

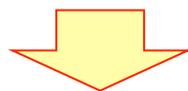


Wouter Verkerke, NIKHEF

An excursion – Collaborative analyses with workspaces

- *How can you reparametrize existing Higgs likelihoods in practice?*
- Write functions expressions corresponding to new parameterization

$$\sigma(gg \rightarrow H) * \text{BR}(H \rightarrow \gamma\gamma) \sim \frac{\kappa_F^2 \cdot \kappa_V^2(\kappa_F, \kappa_V)}{0.75 \cdot \kappa_F^2 + 0.25 \cdot \kappa_V^2}$$



```
w.factory("expr::mu_gg_func(' (KF2*Kg2)/  
                                (0.75*KF2+0.25*KV2) ',  
                                KF2, Kg2, KV2) );
```

- Import transformation in workspace, edit *existing* model

```
w.import(mu_gg_func) ;  
w.factory("EDIT::newmodel(model, mu_gg=mu_gg_gunc)") ;
```

HistFactory

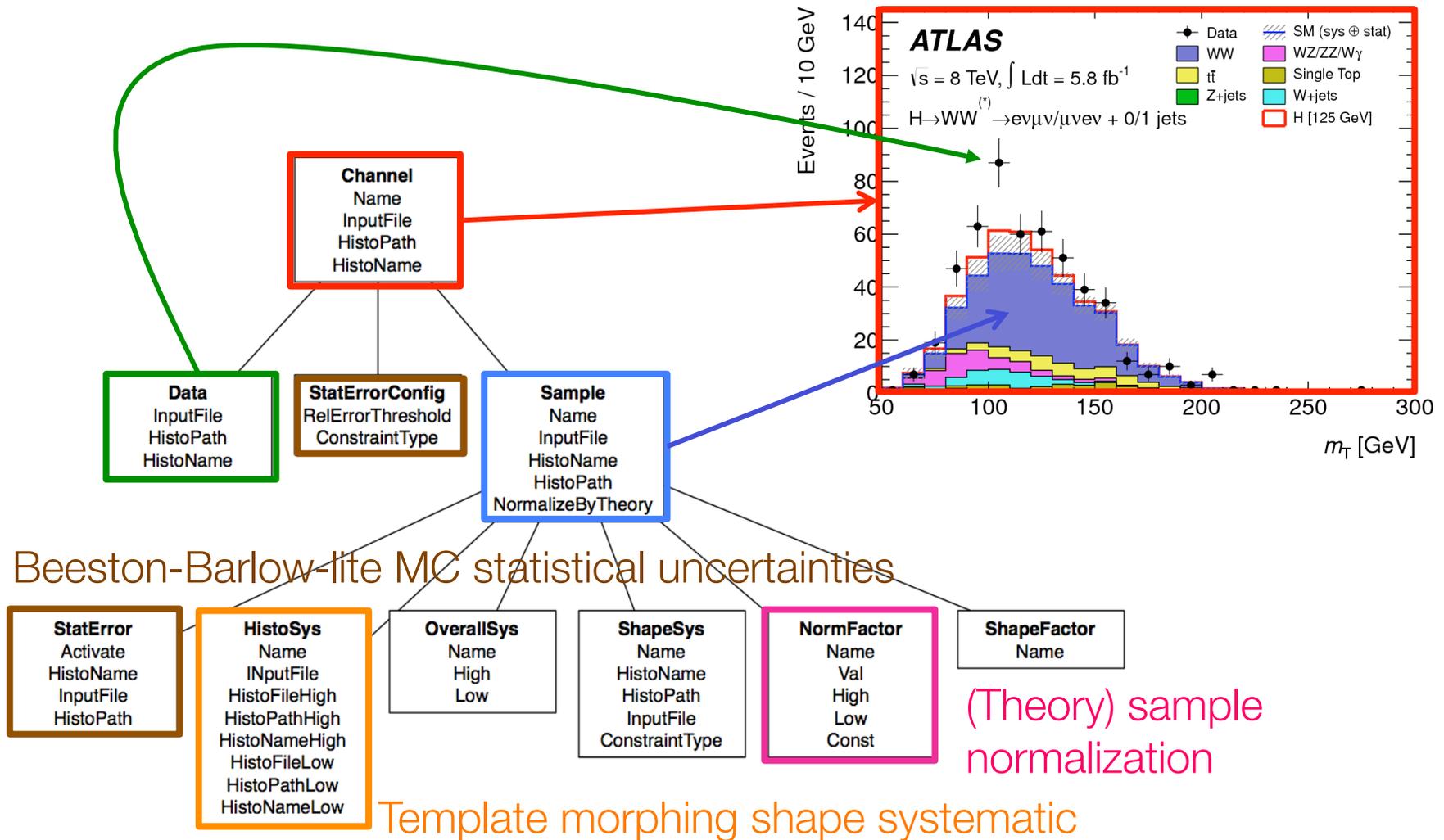
K. Cranmer, A. Shibata, G. Lewis, L. Moneta, W. Verkerke (2010)

HistFactory – structured building of binned template models

- **RooFit modeling building blocks** allow to easily construct likelihood models that model shape and rate systematics with one or more nuisance parameter
 - Only few lines of code per construction
- Typical LHC analysis required modeling of 10-50 systematic uncertainties in $O(10)$ samples in anywhere between 2 and 100 channels → **Need structured formalism to piece together model from specifications.** This is the purpose of HistFactory
- **HistFactory conceptually similar to workspace factory**, but has much higher level semantics
 - Elements represent physics concepts (channels, samples, uncertainties and their relation) rather than mathematical concepts
 - Descriptive elements are represented by C++ objects (like roofit), and can be configured in C++, or alternatively from an XML file
- **HistFactory builds a RooFit (mathematical) model from a physics model.**

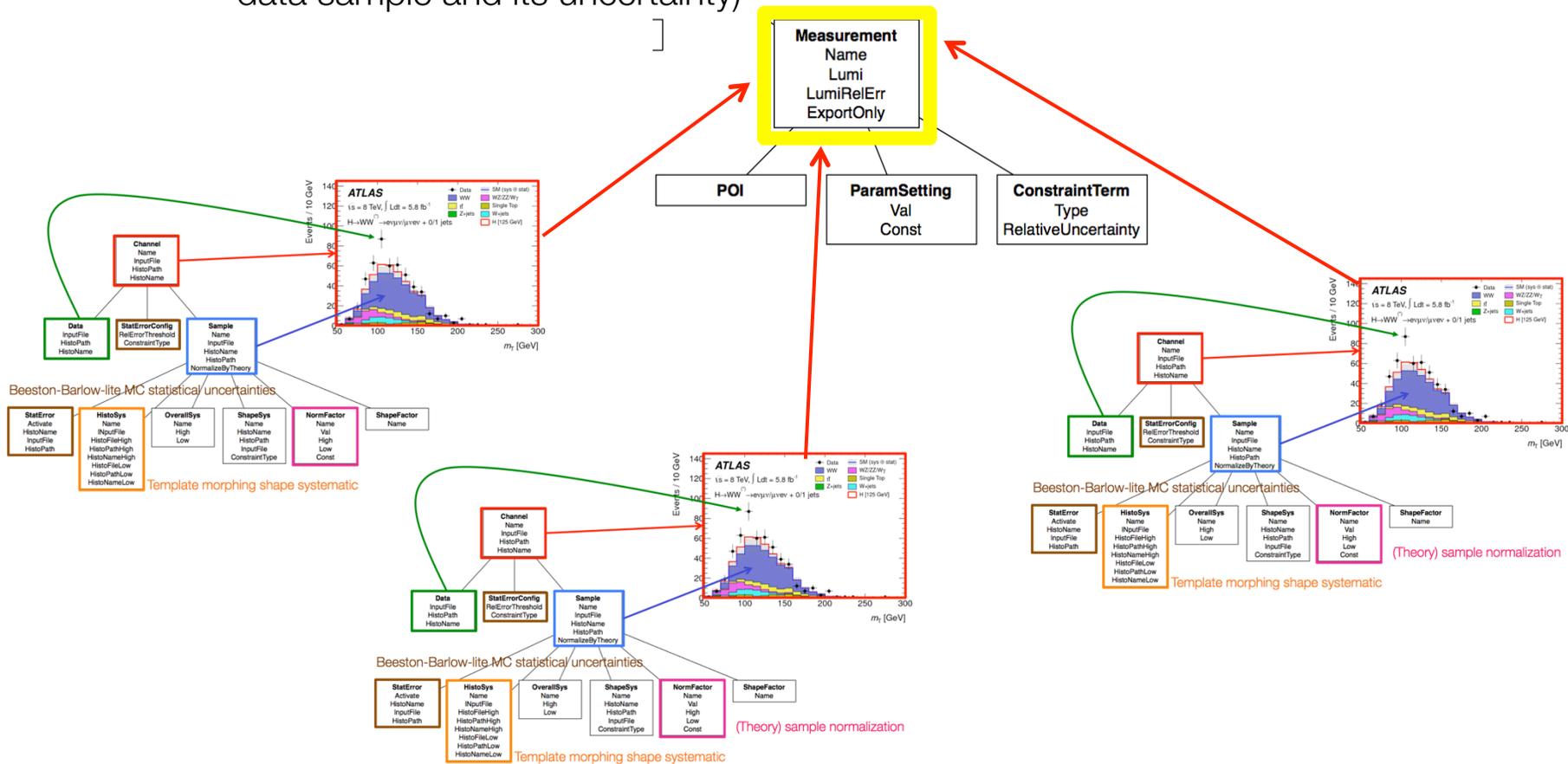
HistFactory elements of a channel

- Hierarchy of concepts for description of one measurement channel



HistFactory elements of measurement

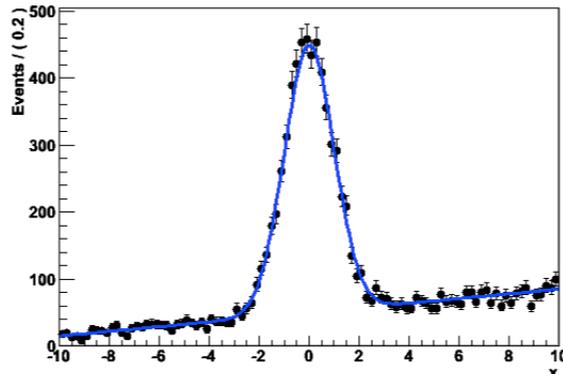
- One or more **channels** are combined to form a **measurement**
 - Along with some extra information (declaration of the POI, the luminosity of the data sample and its uncertainty)



Once physics model is defined, one line of code will turn it into a RooFit likelihood

How is Higgs discovery different from a simple fit?

Gaussian + polynomial



ROOT TH1

ROOT TF1

$$L(\vec{N} | \mu, \vec{\theta}) = \prod_i \text{Poisson}(N_i | f(x_i, \mu, \vec{\theta}))$$

“inside ROOT”



Maximum Likelihood estimation of parameters μ, θ using MINUIT (MIGRAD, HESSE, MINOS)



$$\mu = 5.3 \pm 1.7$$

Likelihood Model
orders of magnitude more complicated. Described

- O(100) signal distributions
- O(10) background sample distr.
- O(10) parameters



representing
syst. uncertainties



$$L(\vec{N}_{ZZ}, \vec{N}_{\tau\tau}, \vec{N}_{WW} | \mu, \vec{\theta}) = \prod \text{Poisson}(N_{ZZ}^i | \dots) \cdot \prod \text{Poisson}(N_{\tau\tau}^i | \dots) \cdot \prod \text{Poisson}(N_{WW}^i | \dots) \dots$$



Frequentist confidence interval construction and/or p-value calculation not available as ‘ready-to-run’ algorithm in ROOT

RooStats

The benefits of modularity

- Perform different statistical test on exactly the same model

RooFit, or RooFit+HistFactory



RooWorkspace



“Simple fit”

(ML Fit with
HESSE or
MINOS)



RooStats
(Frequentist
with toys)



RooStats
(Frequentist
asymptotic)

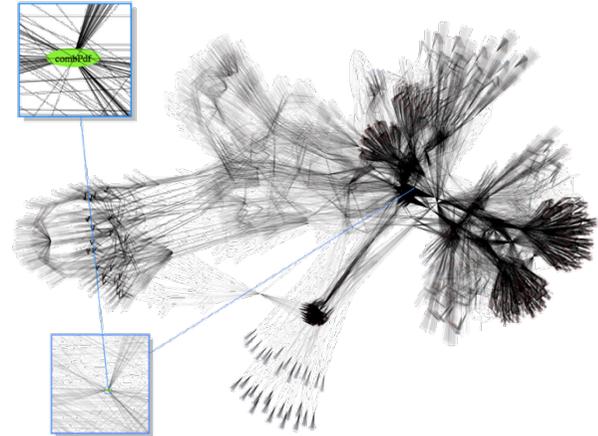


RooStats
Bayesian
MCMC

Maximum Likelihood estimation as simple statistical analysis

- **Step 1** – Construct the likelihood function $L(x|p)$

```
RooWorkspace w("w") ;  
w.factory("Gaussian::sig(x[-10,10],m[0],s[1])") ;  
w.factory("Chebychev::bkg(x,a1[-1,1])") ;  
w.factory("SUM::model(fsig[0,1]*sig,bkg)") ;  
w.writeToFile("L.root") ;
```



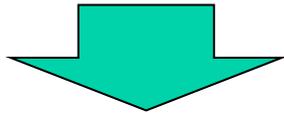
RooWorkspace

- **Step 2** – Statistical tests on parameter of interest p

```
RooWorkspace* w=TFile::Open("L.root")->Get("w") ;  
RooAbsPdf* model = w->pdf("model") ;  
pdf->fitTo(data) ;
```

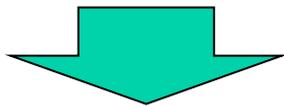
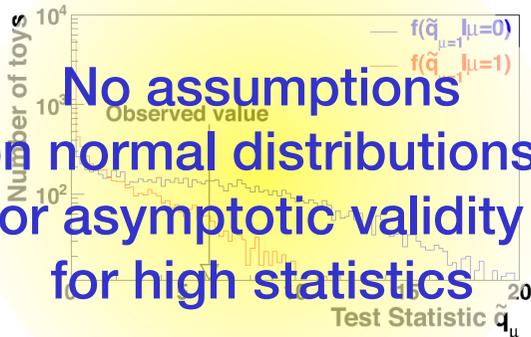
The need for fundamental statistical techniques

Frequentist statistics



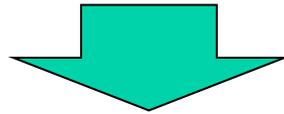
$$\lambda_{\mu}(\vec{N}_{obs}) = \frac{L(\vec{N} | \mu)}{L(\vec{N} | \hat{\mu})}$$

No assumptions on normal distributions, or asymptotic validity for high statistics



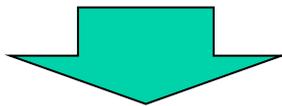
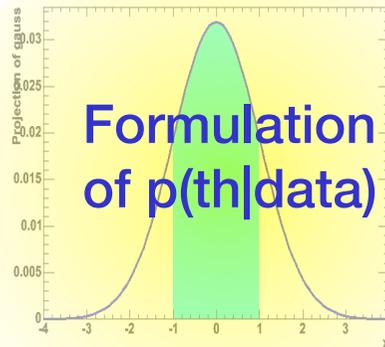
Confidence interval or p-value

Bayesian statistics



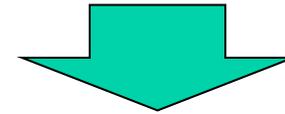
$$P(\mu) \propto L(x | \mu) \cdot \pi(\mu)$$

Formulation of $p(\theta|data)$

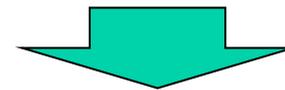
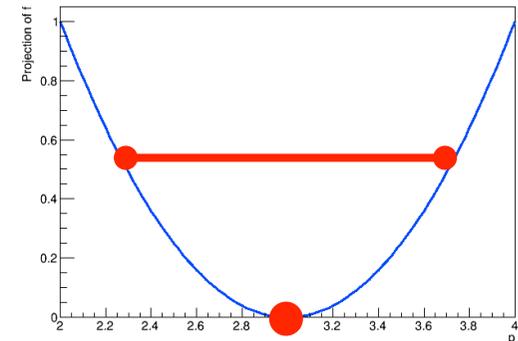


Posterior on s or Bayes factor

Maximum Likelihood



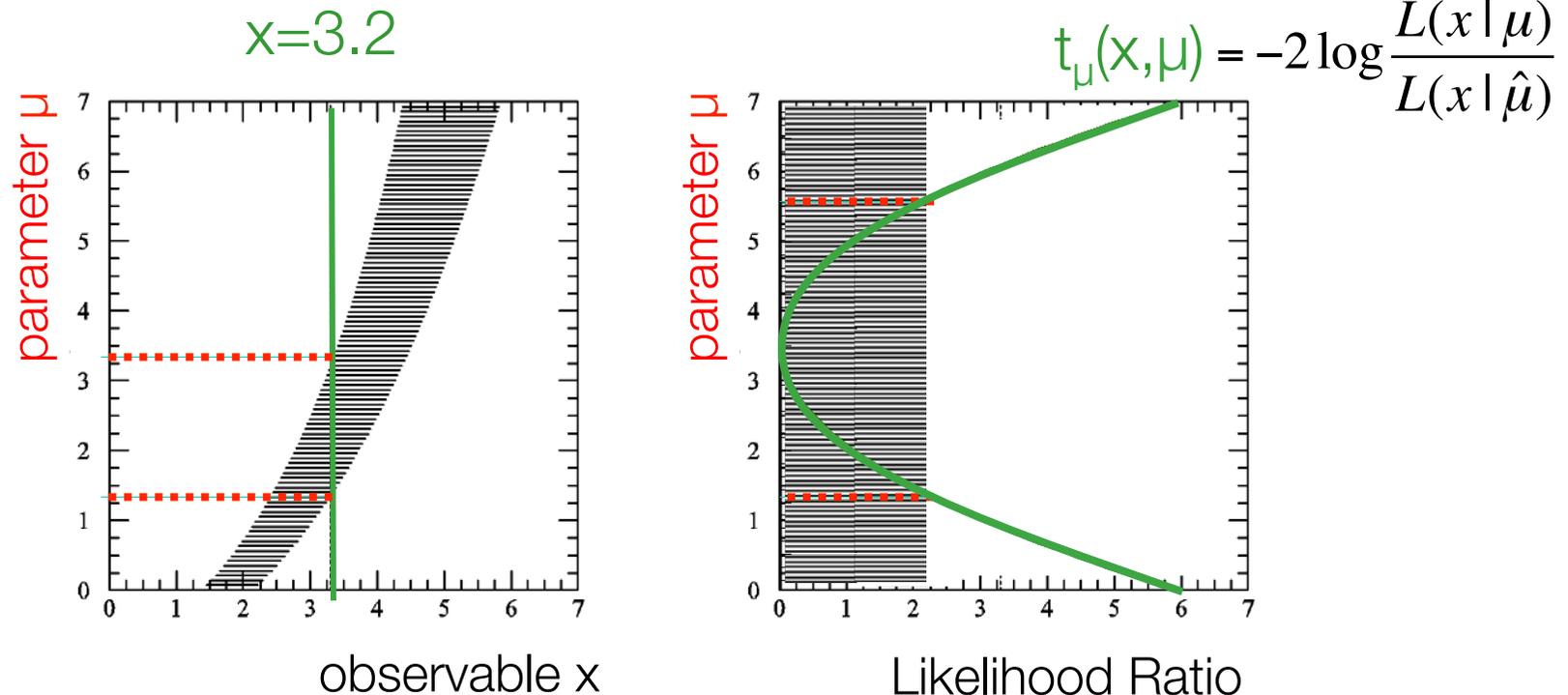
$$\left. \frac{d \ln L(\vec{p})}{d\vec{p}} \right|_{p_i = \hat{p}_i} = 0$$



$s = x \pm y$

But fundamental techniques can be complicated to execute...

- Example of confidence interval calculation with Neyman construction
 - Need to construct ‘confidence belt’ using toy MC. Intersection observed data with belt defined interval in POI with guaranteed coverage



- Expensive, complicated procedure, *but completely procedural once Likelihood and parameter of interest are fixed*
 - Can be wrapped in a tool that runs effectively ‘out-of-the-box’

Running RooStats interval calculations 'out-of-the-box'

- Confidence intervals calculated with model

- 'Simple Fit'

```
RooAbsReal* nll = myModel->createNLL(data) ;  
Roofit m(*nll) ;  
m.migrad() ;  
m.hesse() ;
```

- Feldman Cousins (Frequentist Confidence Interval)

```
FeldmanCousins fc ;  
fc.SetPdf(myModel) ;  
fc.SetData(data) ; fc.SetParameters(myPOU) ;  
fc.UseAdaptiveSampling(true) ;  
fc.FluctuateNumDataEntries(false) ;  
fc.SetNBins(100) ; // number of points to test per parameter  
fc.SetTestSize(.1) ;  
ConfInterval* fcint = fc.GetInterval() ;
```

- Bayesian (MCMC)

```
UniformProposal up ;  
MCMCCalculator mc ;  
mc.SetPdf(w::PC) ;  
mc.SetData(data) ; mc.SetParameters(s) ;  
mc.SetProposalFunction(up) ;  
mc.SetNumIters(100000) ; // steps in the chain  
mc.SetTestSize(.1) ; // 90% CL  
mc.SetNumBins(50) ; // used in posterior histogram  
mc.SetNumBurnInSteps(40) ;  
ConfInterval* mcmcint = mc.GetInterval() ;
```

But you can also look 'in the box' and build your own

Tool to calculate p-values for a given hypothesis $\int_{q_{\mu,obs}}^{\infty} f(q_{\mu} | \mu') dq_{\mu}$

```
// create first HypoTest calculator (N.B null is s+b model)
FrequentistCalculator fc(*data, *bModel, *sbModel);

// configure ToyMCSampler and set the test statistics
ToyMCSampler *toymcs = (ToyMCSampler*)fc.GetTestStatSampler();

ProfileLikelihoodTestStat profll(*sbModel->GetPdf());
// for CLs (bounded intervals) use one-sided profile likelihood
profll.SetOneSided(true);
toymcs->SetTestStatistic(&profll);

HypoTestInverter calc(*fc);
calc.UseCLs(true);

// configure and run the scan
calc.SetFixedScan(npoints, poimin, poimax);
HypoTestInverterResult * r = calc.GetInterval();

// get result and plot it
double upperLimit = r->UpperLimit();
double expectedLimit = r->GetExpectedUpperLimit(0);

HypoTestInverterPlot *plot = new HypoTestInverterPlot("hi", "", r);
plot->Draw();
```

$f(q_{\mu} | \mu')$

Tool to construct
test statistic
distribution

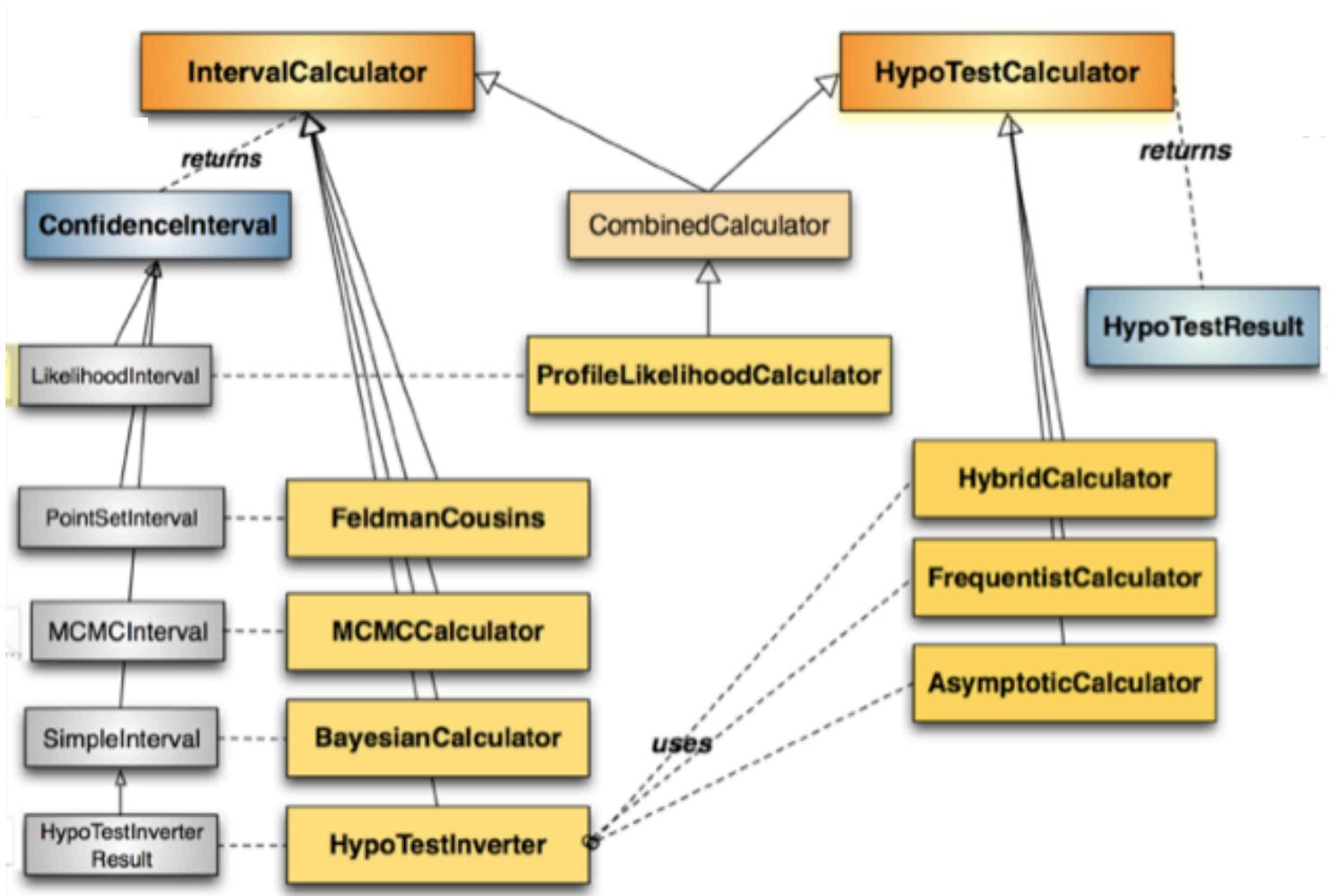
$q_{\mu}(\mu')$

The test statistic
to be used for
the calculation
of p-values

Tool to construct
interval from
hypo test results

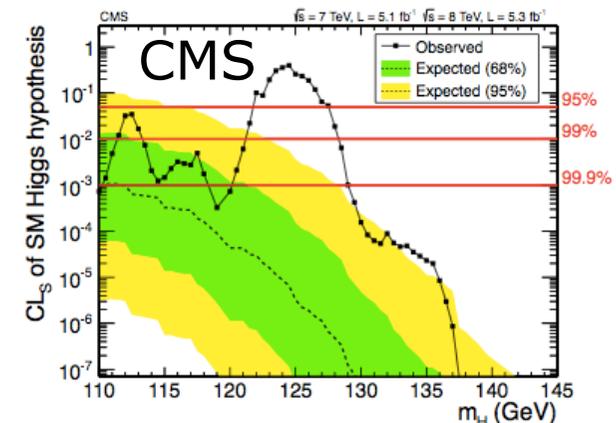
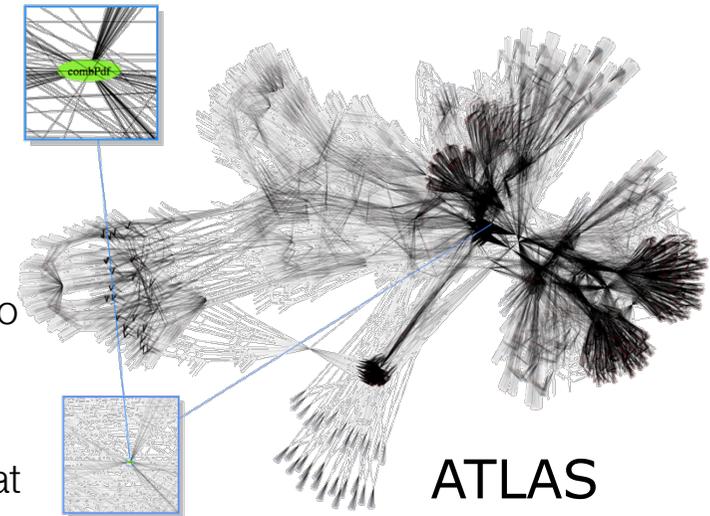
Offset advanced control over details of statistical procedure (use of CLS, choice of test statistic, boundaries...)

RooStats class structure



Summary

- **RooFit** and **RooStats** allow you to perform advanced statistical data analysis
 - LHC Higgs results a prominent example
- **RooFit** provides (almost) limitless model building facilities
 - Concept of persistable model workspace allows to separate model building and model interpretation
 - **HistFactory** package introduces structured model building for binned likelihood template models that are common in LHC analyses
- Concept of RooFit **Workspace** has completely restructured HEP analysis workflow with ‘collaborative modeling’
- **RooStats** provide a wide set of statistical tests that can be performed on RooFit models
 - Bayesian, Frequentist and Likelihood-based test concepts



Modeling 1

**Systematic uncertainties,
“Profile Likelihood”:**

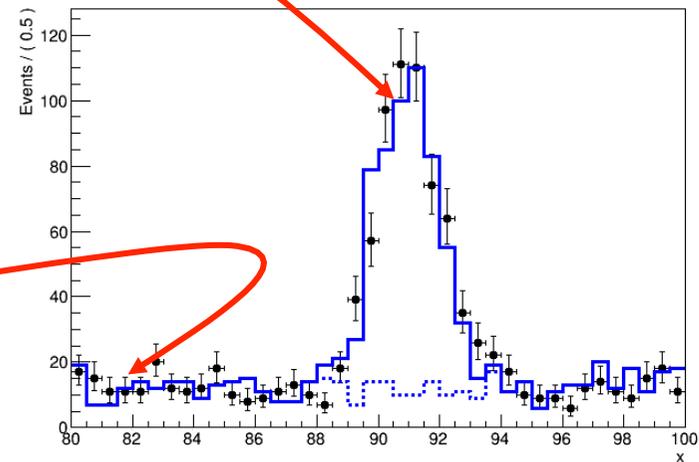
**Counting models
with nuisance parameters**

Most statistics textbooks deal with the *ideal* experiment

- The “only thing” you need to do (as an experimental physicist) is to formulate the likelihood function for your measurement
- For an ideal experiment, where signal and background are assumed to have perfectly known properties, this is trivial

$$L(\vec{N} | \mu) =$$

$$\prod_{bins} Poisson(N_i | \mu \cdot \tilde{s}_i + \tilde{b}_i)$$



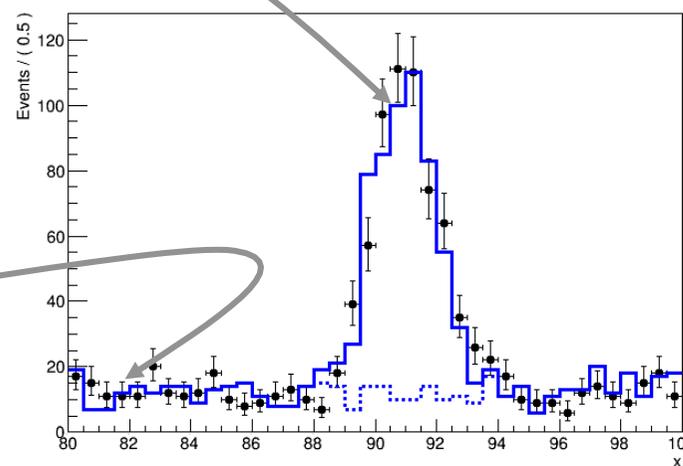
- So far only considered a single parameter in the likelihood: the physics *parameter of interest*, usually denoted as μ

The imperfect experiment

- In realistic measurements many effect that we don't control exactly influence measurements of parameter of interest
- How do you model these uncertainties in the likelihood?

$$L(\vec{N} | \mu) =$$

$$\prod_{bins} Poisson(N_i | \mu \cdot \tilde{s}_i + \tilde{b}_i)$$

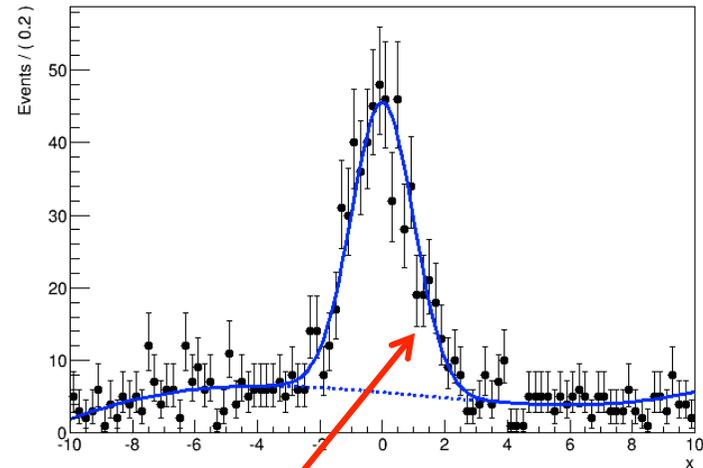
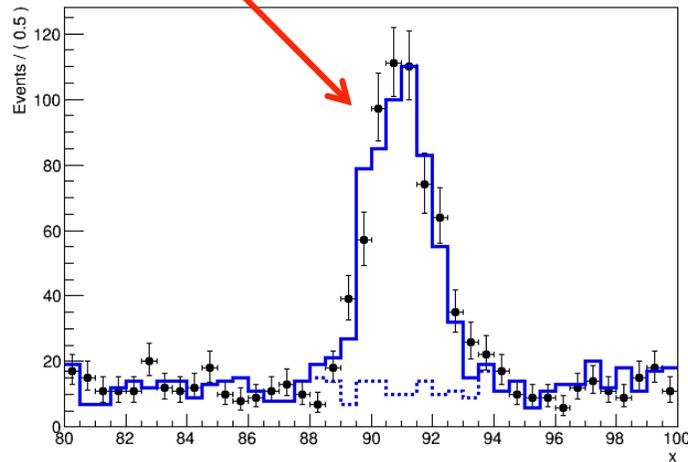


*Signal and background predictions
are affected by (systematic) uncertainties*

Adding parameters to the model

- We can describe uncertainties in our model by adding new parameters of which the value is uncertain

$$L(\vec{N} | \mu) = \prod_{bins} Poisson(N_i | \mu \cdot \tilde{s}_i + \tilde{b}_i)$$

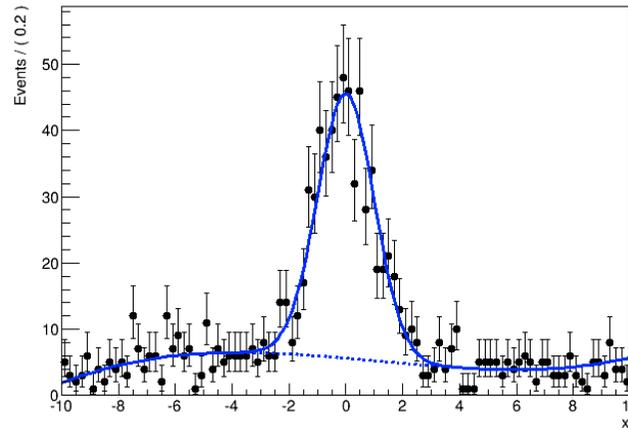


$$L(x | f, m, \sigma, a_0, a_1, a_2) = fG(x, m, \sigma) + (1 - f)Poly(x, a_0, a_1, a_2)$$

- These additional model parameters are not 'of interest', but we need them to model uncertainties → 'Nuisance parameters'

What are the nuisance parameters of your *physics model*?

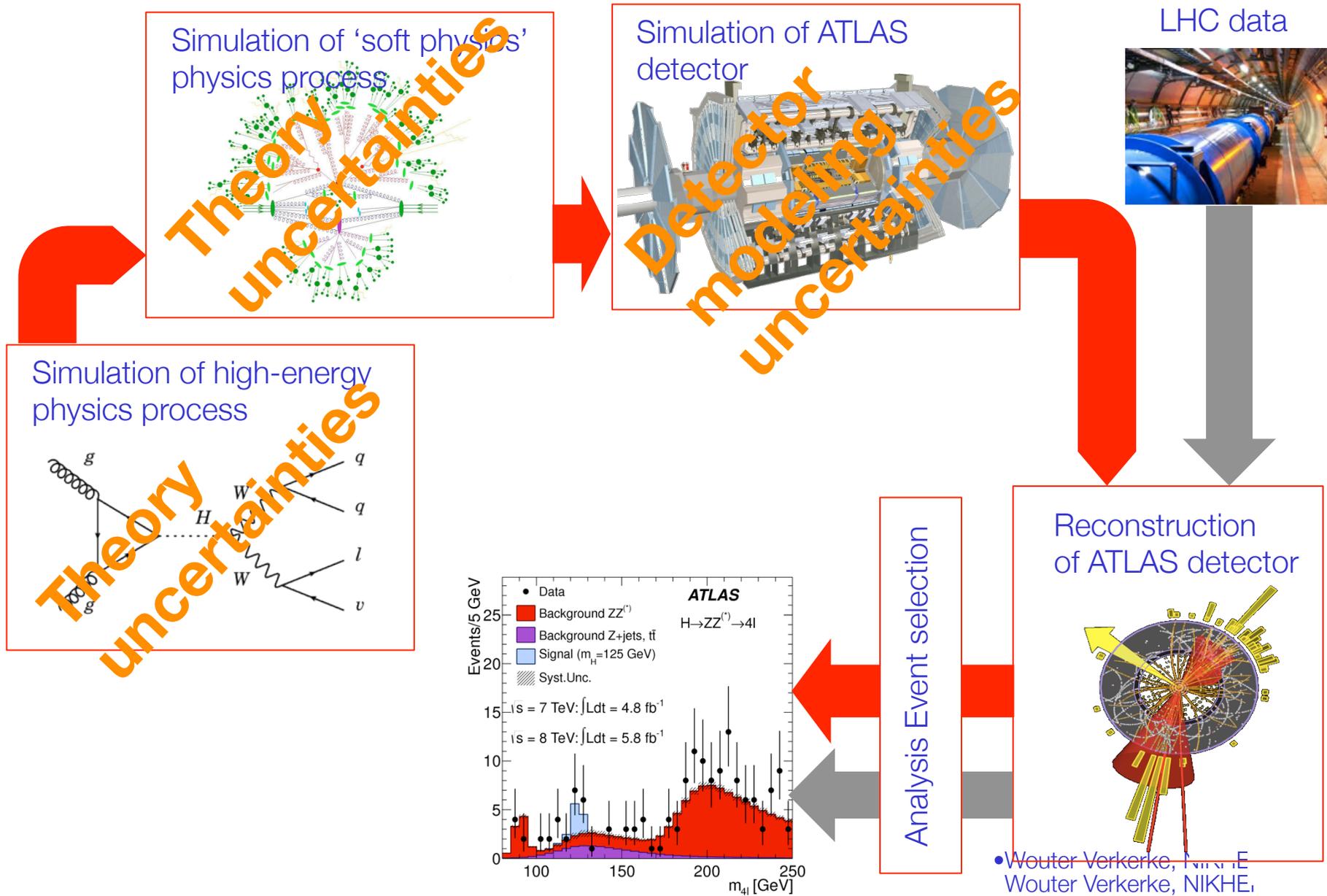
- *Empirical modeling of uncertainties*, e.g. polynomial for background, Gaussian for signal, is easy to do, but may lead to hard questions



$$L(x | f, m, \sigma, a_0, a_1, a_2) = fG(x, m, \sigma) + (1 - f)Poly(x, a_0, a_1, a_2)$$

- *Is your model correct?* (Is true signal distr. captured by a Gaussian?)
- *Is your model flexible enough?* (4th order polynomial, or better 6th?)
- *How do model parameters connect to known detector/theory uncertainties in your distribution?*
 - what conceptual uncertainty do your parameters represent?

The simulation workflow and origin of uncertainties



Typical systematic uncertainties in HEP

- **Detector-simulation related**
 - “The Jet Energy scale uncertainty is 5%”
 - “The b-tagging efficiency uncertainty is 20% for jets with $p_T < 40$ ”
- **Physics/Theory related**
 - The top cross-section uncertainty is 8%
 - “Vary the factorization scale by a factor 0.5 and 2.0 and consider the difference the systematic uncertainty”
 - “Evaluate the effect of using Herwig and Pythia and consider the difference the systematic uncertainty”
- **MC simulation statistical uncertainty**
 - Effect of (bin-by-bin) statistical uncertainties in MC samples

What can you do with *systematic* uncertainties

- As most of the typical systematic prescriptions **have no immediately apparent parametric formulation in your likelihood**, common approach is ‘vary setting, rerun analysis, observe the difference’
- This common ‘naïve’ approach to assess effect of systematic uncertainties amounts to simple error propagation
- Error propagation procedure in a nutshell
 - Make nominal measurement (using your favorite statistical inference procedure)
 - Change setting in detector simulation or theory (e.g. shift Jet Calibration scale by ‘1 sigma’ up and down) Redo measurement procedure for each shift
 - Consider propagated effect of shifted setting the systematic uncertainty

$$\mu = \underbrace{\mu_{nom} \pm \sigma_{stat}}_{\text{From statistical analysis}} \pm \underbrace{(\mu_{syst}^{up} - \mu_{syst}^{down}) / 2}_{\text{Systematic uncertainty from error propagation}} \pm \dots$$

Pros and cons of the 'naïve' approach

- **Pros**

- It's easy to do
- It results in a seemingly easy-to-interpret table of systematics

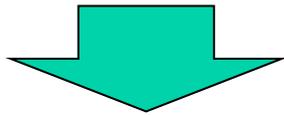
- **Cons**

- Uncorrelated source of systematic uncertainty can have correlated effect on measurement → **Completely ignored**
- Magnitude of stated systematic uncertainty may be incompatible with measurement result → **Completely ignored**
- **You lost the connection with fundamental statistical techniques** (i.e. evaluation of systematic uncertainties is completely detached from statistical procedure used to estimate physics quantity of interest) → **No prescription to make confidence intervals, Bayesian posteriors etc in this way**
- No calibrated probabilistic statements possible (95% C.L.)

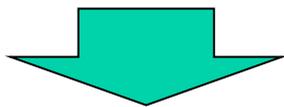
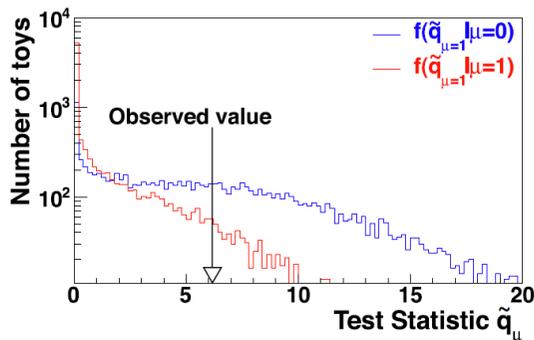
- 'Profiling' → Incorporate a description of systematic uncertainties in the likelihood function that is used in statistical procedures

Everything starts with the likelihood

Frequentist statistics

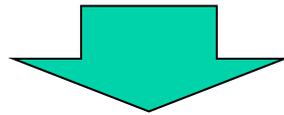


$$\lambda_{\mu}(\vec{N}_{obs}) = \frac{L(\vec{N} | \mu)}{L(\vec{N} | \hat{\mu})}$$

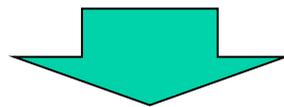
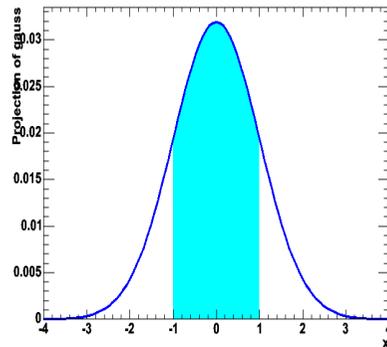


**Confidence interval
or p-value**

Bayesian statistics

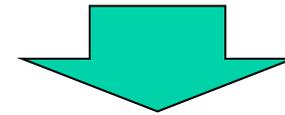


$$P(\mu) \propto L(x | \mu) \cdot \pi(\mu)$$

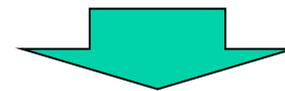
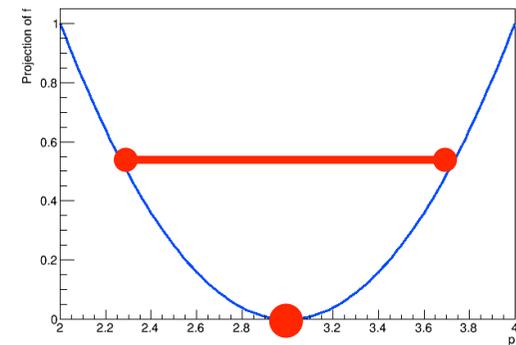


**Posterior on s
or Bayes factor**

Maximum Likelihood



$$\left. \frac{d \ln L(\vec{p})}{d \vec{p}} \right|_{p_i = \hat{p}_i} = 0$$

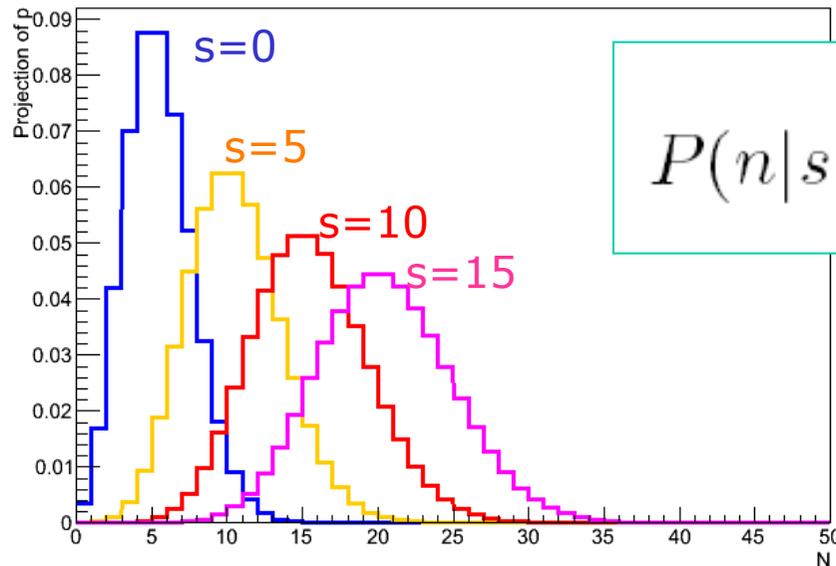


s = x ± y

•Wouter Verkerke, NIKHEF

Introducing uncertainties – a non-systematic example

- The original model (with fixed b)



$$P(n|s + b) = \frac{(s + b)^n}{n!} e^{-(s+b)}$$

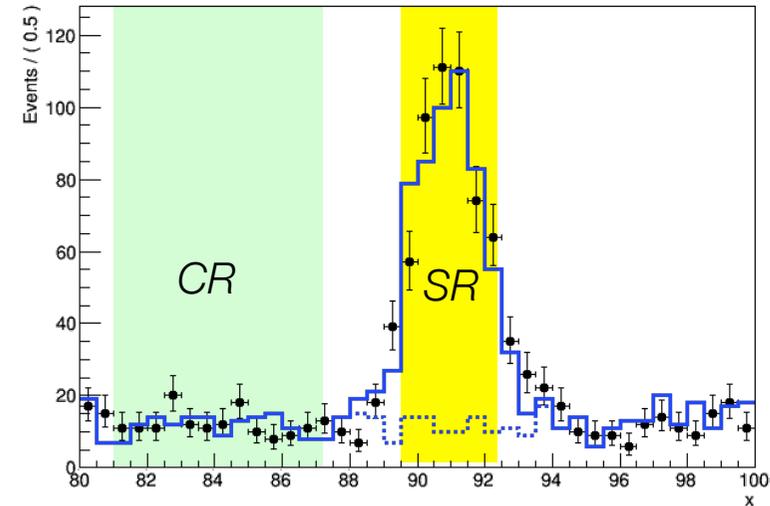
- Now consider b to be uncertain

$$L(N|s) \rightarrow L(N|s,b)$$

- The experimental data contains insufficient to constrain both s and $b \rightarrow$ Need to add an additional measurement to constrain b

The sideband measurement

- Suppose your data in reality looks like this →



Can estimate level of background in the ‘signal region’ from event count in a ‘control region’ elsewhere in phase space

$$L_{SR}(s, b) = \text{Poisson}(N_{SR} | s + b)$$

NB: Define parameter ‘b’ to represent the amount of bkg in the SR.

$$L_{CR}(b) = \text{Poisson}(N_{CR} | \tilde{\tau} \cdot b)$$

Scale factor τ accounts for difference in size between SR and CR

“Background uncertainty constrained from the data”

- Full likelihood of the measurement (‘simultaneous fit’)

$$L_{full}(s, b) = \text{Poisson}(N_{SR} | s + b) \cdot \text{Poisson}(N_{CR} | \tilde{\tau} \cdot b)$$

Generalizing the concept of the sideband measurement

- Background uncertainty from sideband clearly clearly not a ‘systematic uncertainty’

$$L_{full}(s, b) = Poisson(N_{SR} | s + b) \cdot Poisson(N_{CR} | \tilde{\tau} \cdot b)$$

- Now consider scenario where b is not measured from a sideband, but is taken from MC simulation **with an 8% cross-section ‘systematic’ uncertainty**

‘Measured background rate by MC simulation’

$$L_{full}(s, b) = Poisson(N_{SR} | s + b) \cdot Gauss(\tilde{b} | b, 0.08)$$

‘Subsidiary measurement’
of background rate

- *We can model this in the same way, because the cross-section uncertainty is also (ultimately) the result of a measurement*

Generalize: ‘sideband’ → ‘subsidiary measurement’

What is a systematic uncertainty?

- Concept & definitions of ‘systematic uncertainties’ originates from physics, not from fundamental statistical methodology.
 - E.g. Glen Cowans (excellent) 198pp book “statistical data analysis” does not discuss systematic uncertainties at all
- A common definition is
 - “Systematic uncertainties are all uncertainties that are not directly due to the statistics of the data”
- But the notion of ‘the data’ is a key source of ambiguity:
 - does it include control measurements?
 - does it include measurements that were used to perform basic (energy scale) calibrations?

Typical systematic uncertainties in HEP

- **Detector-simulation related**

- “The Jet Energy scale uncertainty is 5%”
- “The b-tagging efficiency uncertainty is 20% for jets with $p_T < 40$ ”

Subsidiary measurement is an actual measurement
→ conceptually similar to a ‘sideband’ fit

- **Physics/Theory related**

- The top cross-section uncertainty is 8%
- “Vary the factorization scale by a factor 0.5 and 2.0 and consider the difference the systematic uncertainty”
- “Evaluate the effect of using Herwig and Pythia and consider the difference the systematic uncertainty”

Subsidiary measurement unclear, but origin of prescription may well be another measurement (if yes, like sideband, if no, what is source of info?)

- **MC simulation statistical uncertainty**

- Effect of (bin-by-bin) statistical uncertainties in MC samples

Subsidiary measurement is a Poisson counting experiment (but now in MC events), otherwise conceptually identical to a ‘sideband fit’

Typical systematic uncertainties in HEP

- **Detector-simulation related**

- “The Jet Energy scale uncertainty is 5%”
- “The b-tagging efficiency uncertainty is 20%”

Subsidiary measurement
is an actual measurement
→ conceptually to

- **P**

Almost all systematic uncertainties are similar in nature to ‘sidebands’ measurements of some form or shape

→ Can always model systematics like sidebands in the Likelihood

And even when they are not the (in)direct result of some measurement (certainty theory uncertainties) we can still model them in that form

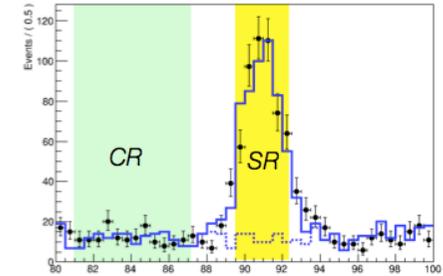
- **MC simulation statistical uncertainty**

- Effect of (bin-by-bin) statistical uncertainties in MC samples

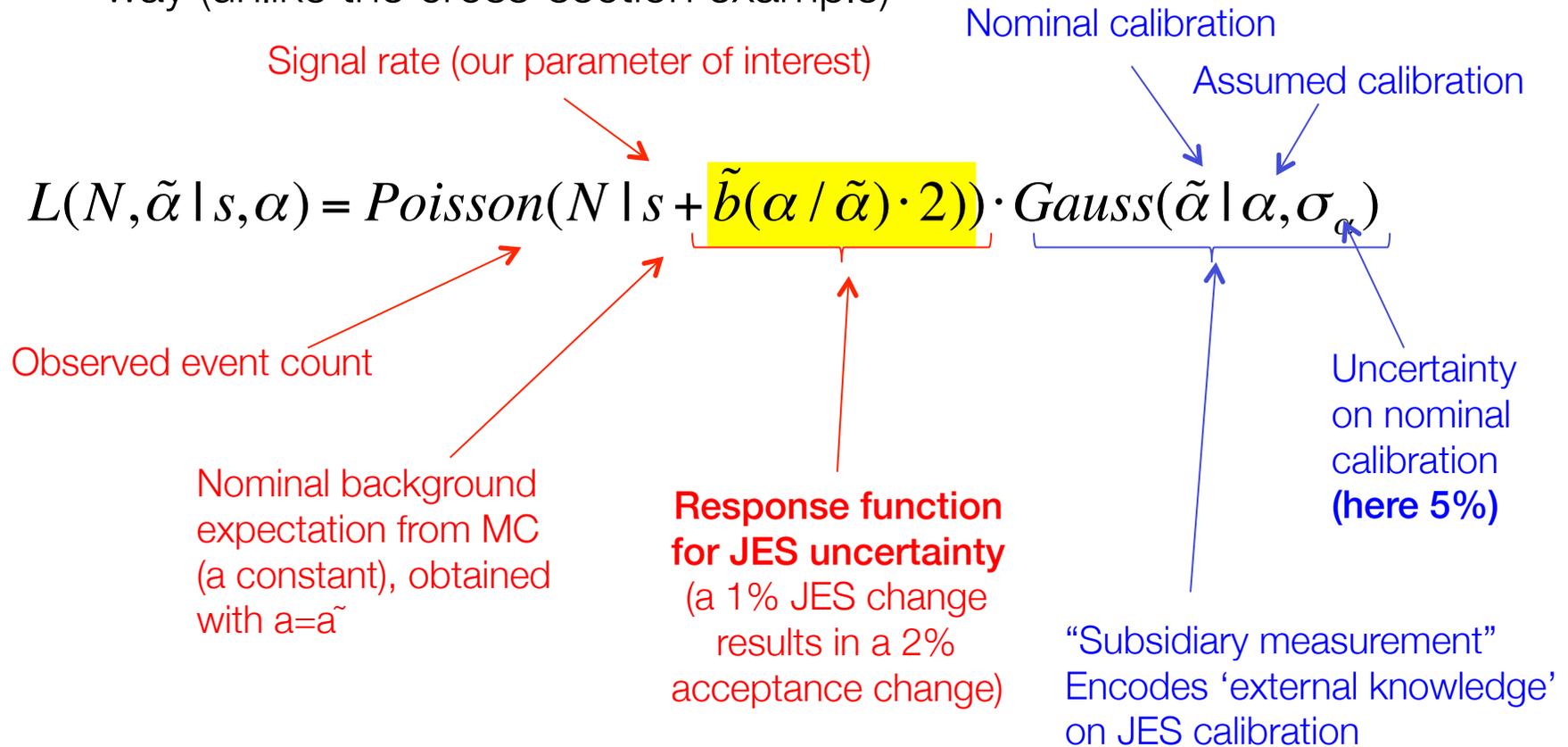
Subsidiary measurement is a Poisson counting experiment (but now in MC events), otherwise conceptually identical to a ‘sideband fit’

Modeling a detector calibration uncertainty

$$L_{full}(s, b) = \text{Poisson}(N_{SR} | s + b) \cdot \text{Gauss}(\tilde{b} | b, 0.08)$$

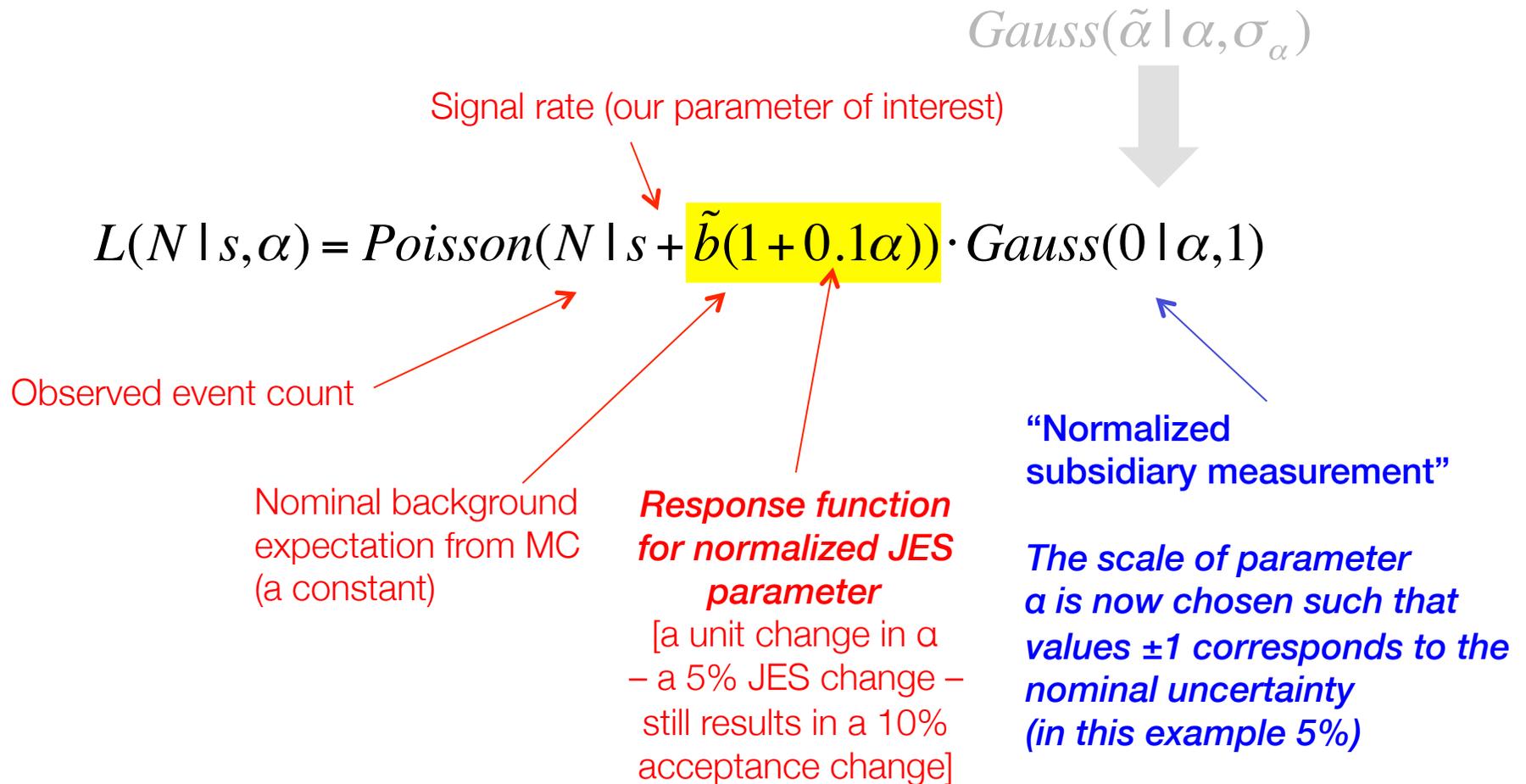


- **Now consider a detector uncertainty**, e.g. jet energy scale calibration, which can affect the analysis acceptance in a non-trivial way (unlike the cross-section example)



Modeling a detector calibration uncertainty

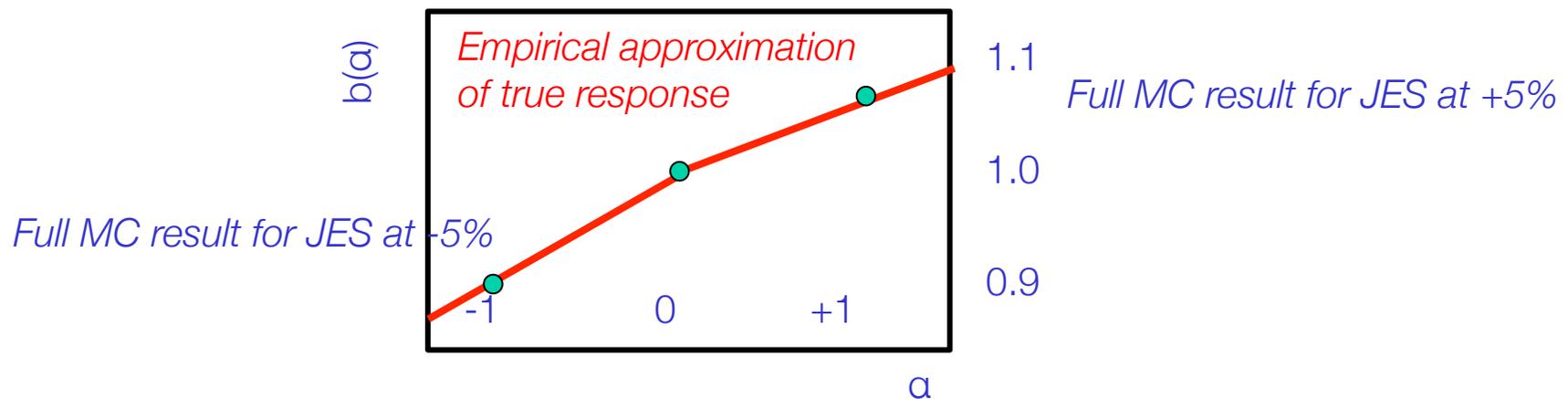
- Simplify expression by renormalizing “subsidiary measurement”



The response function as empirical model of full simulation

$$L(N, 0 | s, \alpha) = \text{Poisson}(N | s + \underbrace{b(\alpha)}) \cdot \text{Gauss}(0 | \alpha, 1)$$

- Note that the response function is generally not linear, but can in principle *always be determined by your full simulation chain*
 - But you cannot run your full simulation chain for any arbitrary ‘systematic uncertainty variation’ → Too much time consuming
 - Typically, run full MC chain for nominal and $\pm 1\sigma$ variation of systematic uncertainty, and approximate response for other values of NP with interpolation
 - For example run at nominal JES and with JES shifted up and down by $\pm 5\%$



What is a systematic uncertainty?

- It is an uncertainty in the Likelihood of your physics measurement that is characterized deterministically, up to a set of parameters, of which the true value is unknown.
- A fully specified systematic uncertainty defines
 - 1: A set of one or more parameters of which the true value is unknown,
 - 2: A response model that describes the effect of those parameters on the measurement (*sampled from full simulation, and interpolation*)
 - 3: A subsidiary measurement of the parameters that constrains the values the parameters can take (implies a specific distribution: Gaussian (*default, CLT*), Poisson (*low-stats counting*), or otherwise)

Names and conventions – ‘profiling’ & ‘constraints’

- The full likelihood function of the form

$$L(N, 0 | s, \alpha) = \text{Poisson}(N | s + b(\alpha)) \cdot \text{Gauss}(0 | \alpha, 1)$$

is usually referred to by physicists as a ‘**profile likelihood**’, and systematics are said to be ‘**profiled**’ when incorporated this way

– Note: statisticians use the word profiling for something else

- Physicists often refer to the **subsidiary measurement** as a ‘**constraint term**’
 - This is correct in the sense that it constrains the parameter α , but this labeling commonly lead to mistaken statements (e.g. that it is a pdf for α)
 - But it is *not* a pdf in the NP

~~$\text{Gauss}(\alpha | 0, 1)$~~

$\text{Gauss}(0 | \alpha, 1)$

Names and conventions

- The ‘subsidiary measurement’ as simplified form of the ‘full calibration measurement’ also illustrates another important point
 - The full likelihood is simply a *joint likelihood of a physics measurement and a calibration measurement* where both terms are treated on equal footing in the statistical procedure
 - In a perfect world, not bound by technical modelling constraints you would use this likelihood

$$L(N, \vec{y} | s, \alpha) = \text{Poisson}(N | s + b(1 + 0.1\alpha)) \cdot L_{JES}(\vec{y} | \alpha, \vec{\theta})$$

where L_{JES} is the full calibration measurement as performed by the Jet calibration group, based on a dataset y , and which may have other parameters θ specific to the calibration measurement.

- Since we are bound by technical constraints, we substitute L_{JES} with simplified (Gaussian) form, but the statistical treatment and interpretation remains the same

MC statistical uncertainties as systematic uncertainty

- Another example of modeling a systematic uncertainty:
MC statistical uncertainty
- Follow same procedure again as before:
 - Define response function (this is trivial for MC statistics:
it is the luminosity ratio of the MC sample and the data sample)
 - Define distribution for the ‘subsidiary measurement’ – This is a Poisson distribution – since MC simulation is also a Poisson process
 - Construct full likelihood (‘profile likelihood’)

$$L(N, N_{MC} | s, b) = \text{Poisson}(N | s + b) \cdot \text{Poisson}(N_{MC} | \tau \cdot b)$$

Constant factor $\tau = L(\text{MC})/L(\text{data})$ 

- Note uncanny similarity to full likelihood of a sideband measurement!

$$L(N, N_{ctl} | s, b) = \text{Poisson}(N | s + b) \cdot \text{Poisson}(N_{ctl} | \tau \cdot b)$$

Modeling multiple systematic uncertainties

- Introduction of multiple systematic uncertainties presents no special issues
- Example JES uncertainty plus generator ISR uncertainty

$$L(N, 0 | s, \alpha_{JES}, \alpha_{ISR}) = P(N | s + \underbrace{b(1 + 0.1\alpha_{JES} + 0.05\alpha_{ISR})}_{\text{Joint response function for both systematics}}) \cdot \underbrace{G(0 | \alpha_{JES}, 1)}_{\text{One subsidiary measurement for each source of uncertainty}} \cdot \underbrace{G(0 | \alpha_{ISR}, 1)}_{\text{One subsidiary measurement for each source of uncertainty}}$$

- A brief note on correlations
 - Word “correlations” often used sloppily – **proper way is to think of correlations of parameter estimators**. Likelihood defines parameters $\alpha_{JES}, \alpha_{ISR}$. The (ML) estimates of these are denoted $\hat{\alpha}_{JES}, \hat{\alpha}_{ISR}$
 - The ML estimators of $\hat{\alpha}_{JES}, \hat{\alpha}_{ISR}$ using the Likelihood of the subsidiary measurements are uncorrelated (since the product factorize in this example)
 - The ML estimators of $\hat{\alpha}_{JES}, \hat{\alpha}_{ISR}$ using the full Likelihood may be correlated. This is due to physics modeling effects encoded in the joint response function

Modeling systematic uncertainties in multiple channels

- Systematic effects that affect multiple measurements should be modeled coherently.
 - Example – Likelihood of two Poisson counting measurements

$$L(N_A, N_B | s, \alpha_{JES}) = P(N_A | s \cdot f_A + b_A \underbrace{(1 + 0.1\alpha_{JES})}_{\substack{\text{JES response} \\ \text{function for} \\ \text{channel A}}}) \cdot P(N_B | s \cdot f_B + b_B \underbrace{(1 - 0.3\alpha_{JES})}_{\substack{\text{JES response} \\ \text{function for} \\ \text{channel B}}}) \cdot \underbrace{G(0 | \alpha_{JES}, 1)}_{\substack{\text{JES} \\ \text{subsidiary} \\ \text{measurement}}}$$

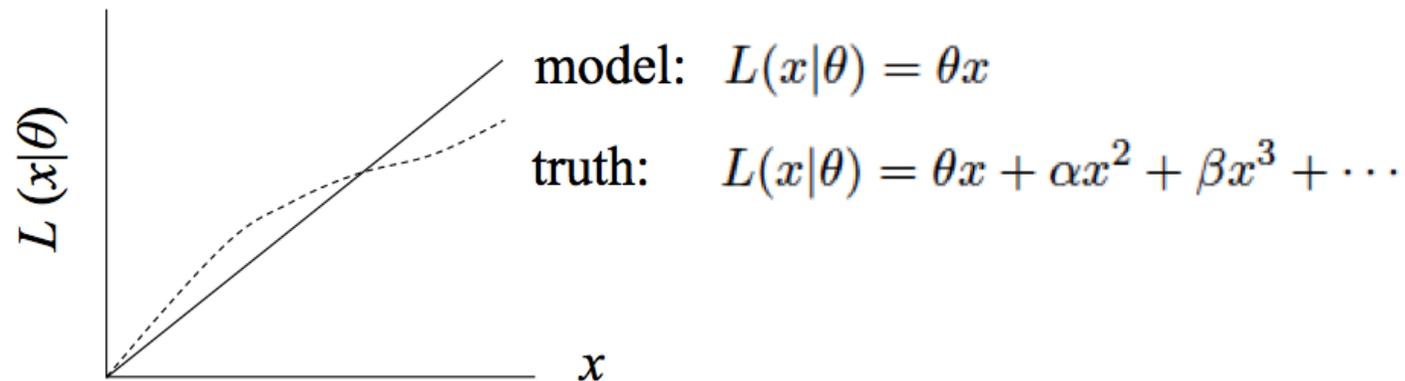
- Effect of changing JES parameter α_{JES} coherently affects both measurement.
- Magnitude and sign effect does not need to be same, this is dictated by the physics of the measurement

Statistics 1

**treatment of nuisance
parameters in statistics
inference**

The statisticians view on nuisance parameters

- In general, our model of the data is not perfect



- Can improve modeling by including additional adjustable parameters
- Goal: some point in the parameter space of the enlarged model should be “true”
- Presence of nuisance parameters decreases the sensitivity of the analysis of the parameter(s) of interest

Treatment of nuisance parameters in variance estimation

- Maximum likelihood estimator of parameter variance is based on 2nd derivative of Likelihood
 - For multi-parameter problems this 2nd derivative is generalized by the **Hessian Matrix** of partial second derivatives

$$\hat{\sigma}(p)^2 = \hat{V}(p) = \left(\frac{d^2 \ln L}{d^2 p} \right)^{-1} \quad \rightarrow \quad \hat{\sigma}(p_i)^2 = \hat{V}(p_{ii}) = (H^{-1})_{ii}$$

$$H(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

- For multi-parameter likelihoods estimate of **covariance** V_{ij} of pair of 2 parameters in addition to variance of individual parameters
 - Usually re-expressed in terms dimensionless correlation coefficients ρ

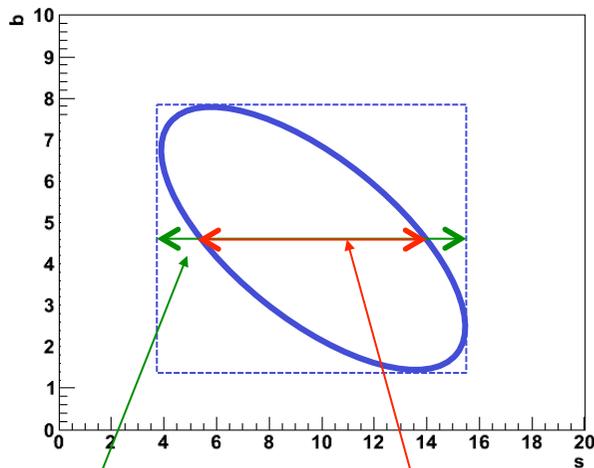
$$V_{ij} = \rho_{ij} \sqrt{V_{ii} V_{jj}}$$

Treatment of nuisance parameters in variance estimation

- Effect of NPs on variance estimates visualized

Scenario 1

Estimators of
POI and NP correlated
i.e. $\rho(s,b) \neq 0$

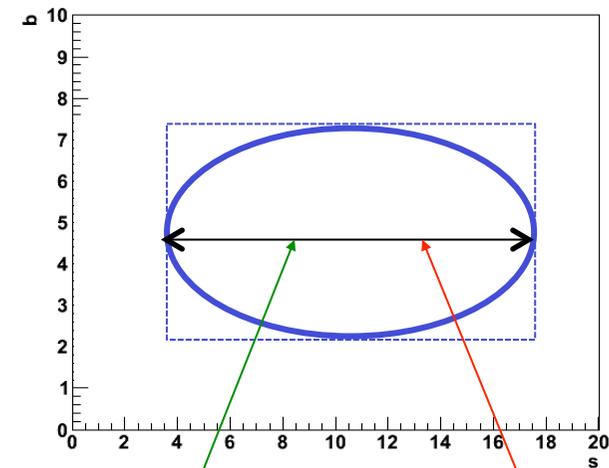


$$\hat{V}(s) \text{ from } \begin{bmatrix} \frac{\partial^2 L}{\partial s^2} & \frac{\partial^2 L}{\partial s \partial b} \\ \frac{\partial^2 L}{\partial s \partial b} & \frac{\partial^2 L}{\partial b^2} \end{bmatrix}^{-1}$$

$$\hat{V}(s) \text{ from } \left[\frac{\partial^2 L}{\partial s^2} \right]_{b=\hat{b}}^{-1}$$

Scenario 2

Estimators of
POI and NP correlated
i.e. $\rho(s,b) = 0$



$$\hat{V}(s) \text{ from } \begin{bmatrix} \frac{\partial^2 L}{\partial s^2} & \frac{\partial^2 L}{\partial s \partial b} \\ \frac{\partial^2 L}{\partial s \partial b} & \frac{\partial^2 L}{\partial b^2} \end{bmatrix}^{-1}$$

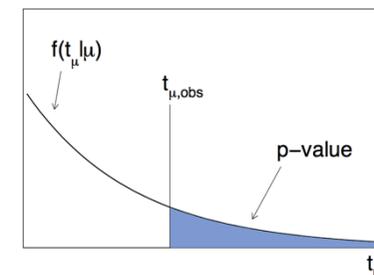
$$\hat{V}(s) \text{ from } \left[\frac{\partial^2 L}{\partial s^2} \right]_{b=\hat{b}}^{-1}$$

Uncertainty on background increases uncertainty on signal

Treatment of NPs in hypothesis testing and conf. intervals

- We've covered frequentist hypothesis testing and interval calculation using likelihood ratios based on a likelihood with a single parameter (of interest) $L(\mu)$
 - Result is p-value on hypothesis with given μ value, or
 - Result is a confidence interval $[\mu_-, \mu_+]$ with values of μ for which p-value is at or above a certain level (the confidence level)
- How do you do this with a likelihood $L(\mu, \theta)$ where θ is a nuisance parameter?
 - With a test statistics q_μ , we calculate p-value for hypothesis θ as

$$p_\mu = \int_{q_{\mu, obs}}^{\infty} f(q_\mu | \mu, \theta) dq_\mu$$



- But what values of θ do we use for $f(q_\mu | \mu, \theta)$?
Fundamentally, we want to reject μ only if $p < \alpha$ for all θ
→ Exact confidence interval

Hypothesis testing & conf. intervals with nuisance parameters

- The goal is that the parameter of interest should be covered at the stated confidence **for every value of the nuisance parameter**
- if there is *any value* of the nuisance parameter which makes the data consistent with the parameter of interest, that value of the POI should be considered:
 - e.g. don't claim discovery if any background scenario is compatible with data
- But: technically very challenging and significant problems with over-coverage
 - Example: **how broadly should 'any background scenario' be defined?** Should we include background scenarios that are clearly incompatible with the observed data?

The profile likelihood construction as compromise

- For LHC the following prescription is used:

$$\text{Given } L(\mu, \theta)$$

↙ NPs
↗ POI

perform hypothesis test for each value of μ (the POI),

using values of nuisance parameter(s) θ that best fit the data under the hypothesis μ

- Introduce the following notation

$$\hat{\theta}(\mu)$$

M.L. estimate of θ for a given value of μ
(i.e. a conditional ML estimate)

- The resulting confidence interval will have exact coverage for the points $(\mu, \hat{\theta}(\mu))$
 - Elsewhere it may overcover or undercover (but this can be checked)

The profile likelihood ratio

- With this prescription we can construct the **profile likelihood ratio** as test statistic

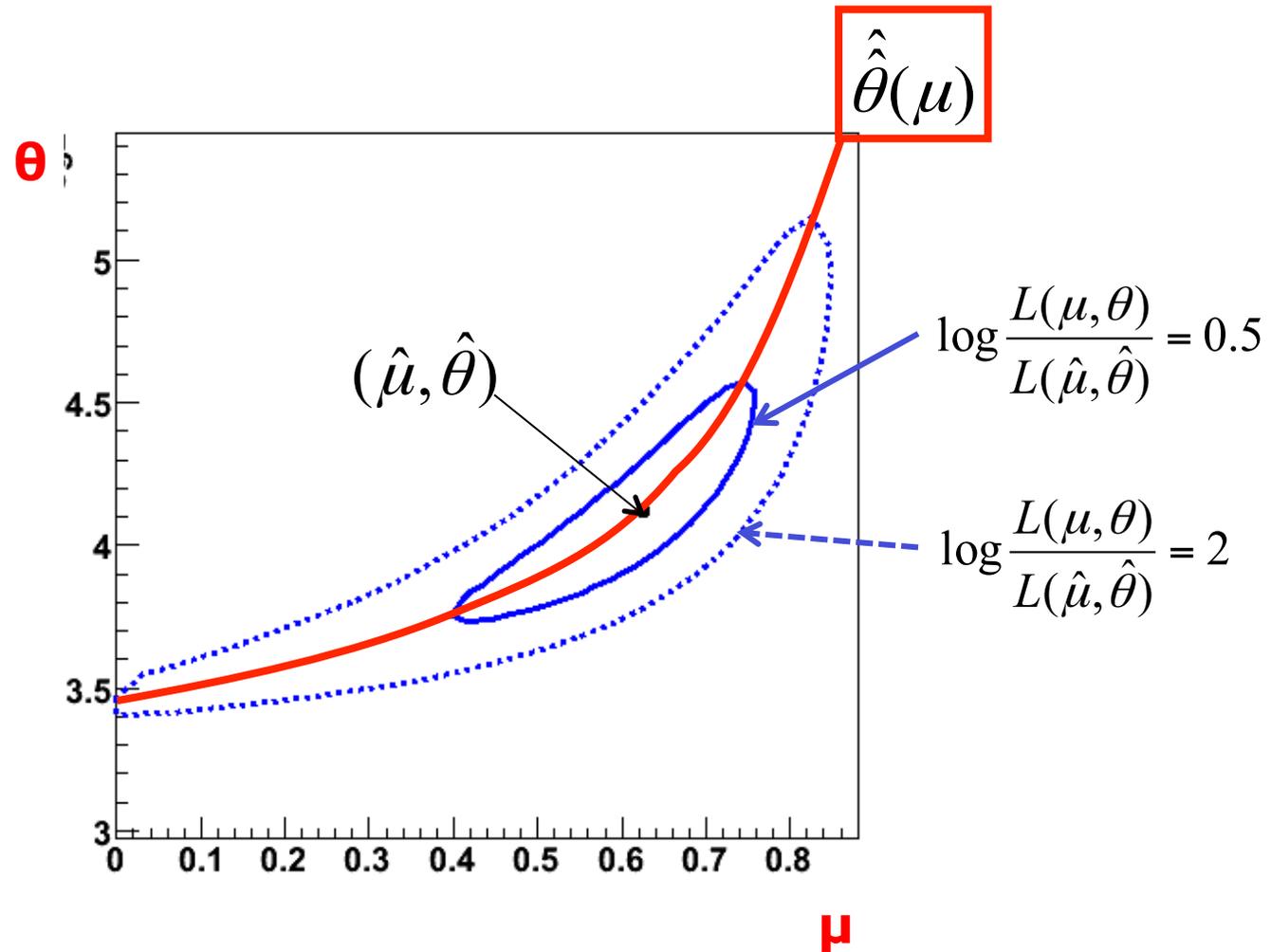
Likelihood for given μ Maximum Likelihood for given μ

$$\lambda(\mu) = \frac{L(\mu)}{L(\hat{\mu})} \quad \rightarrow \quad \lambda(\mu) = \frac{L(\mu, \hat{\theta}(\mu))}{L(\hat{\mu}, \hat{\theta})}$$

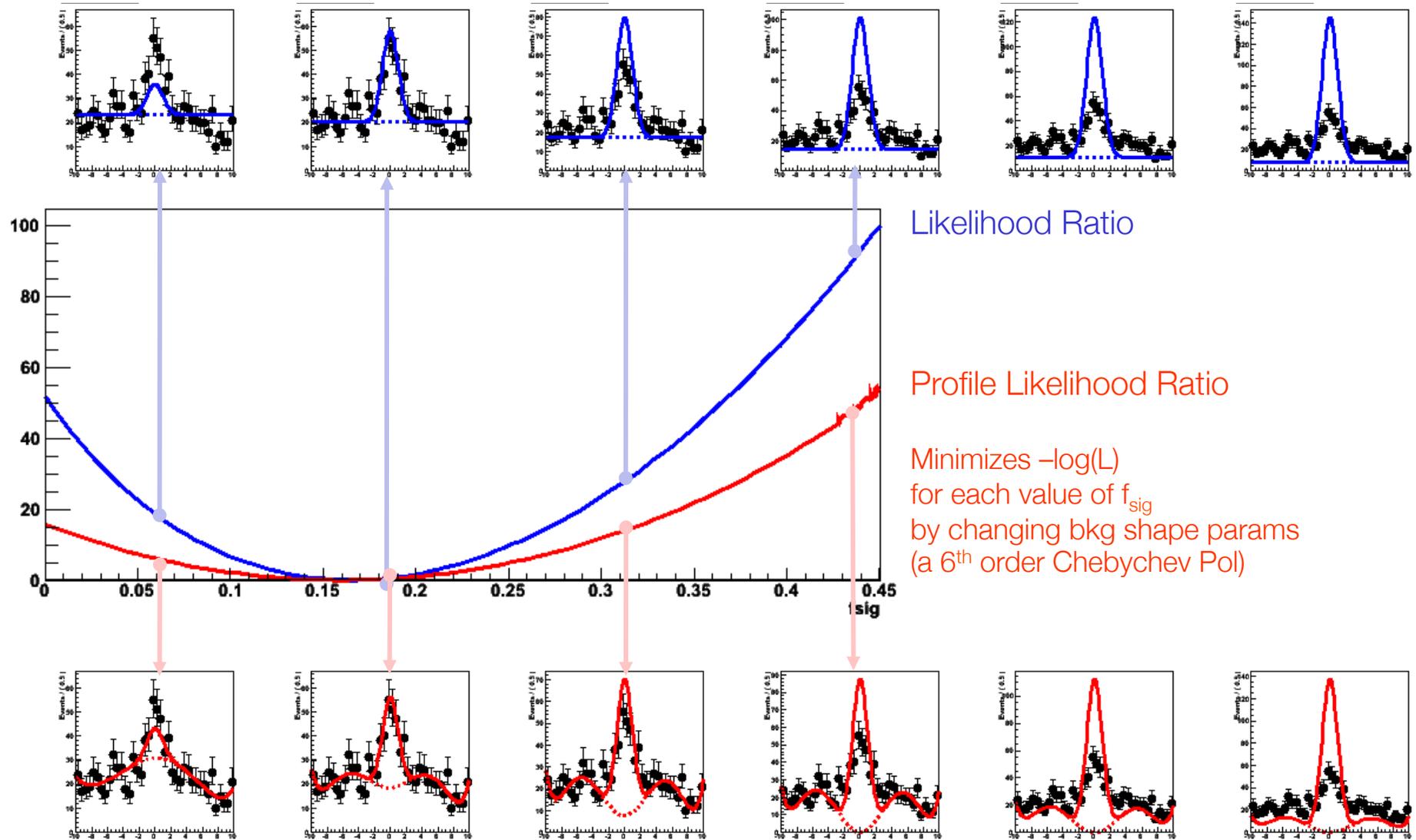
Maximum Likelihood Maximum Likelihood

- NB: value profile likelihood ratio does *not* depend on θ

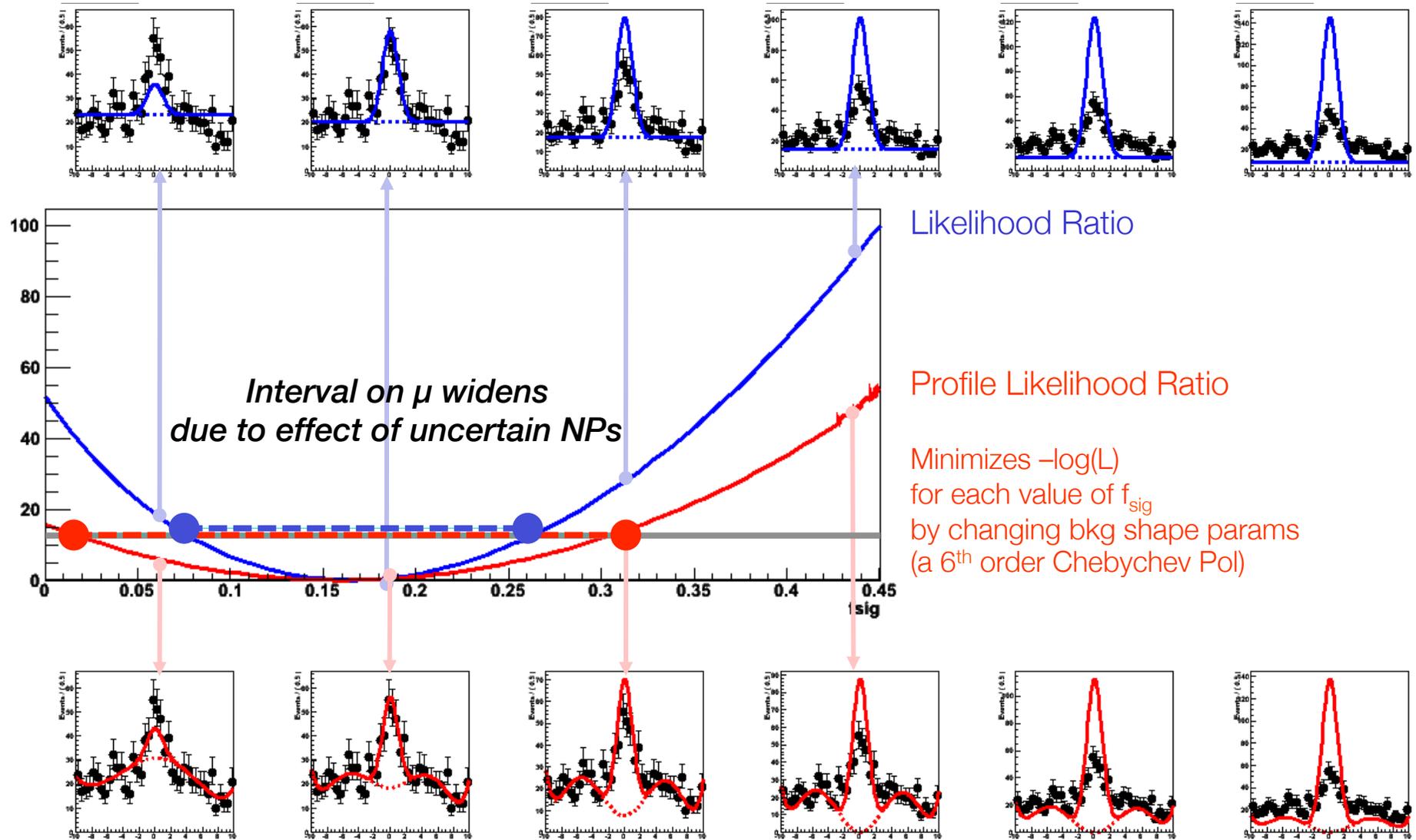
Profiling illustration with one nuisance parameter



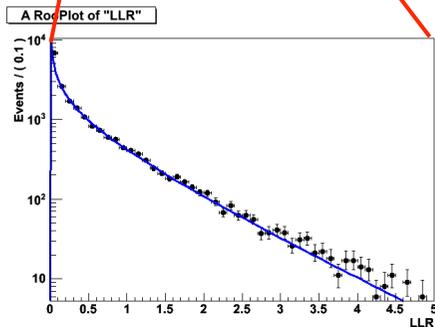
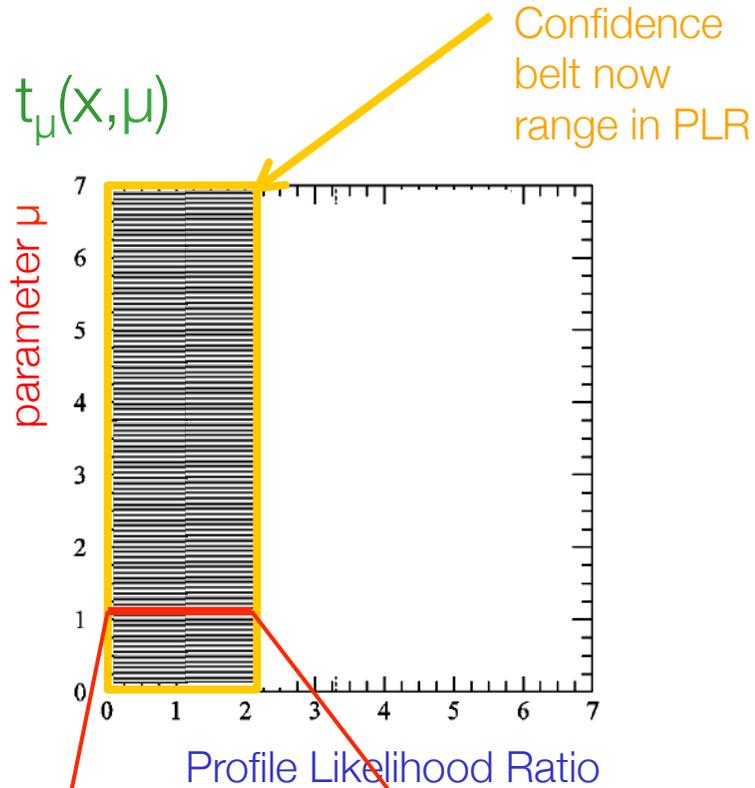
Profile scan of a Gaussian plus Polynomial probability model



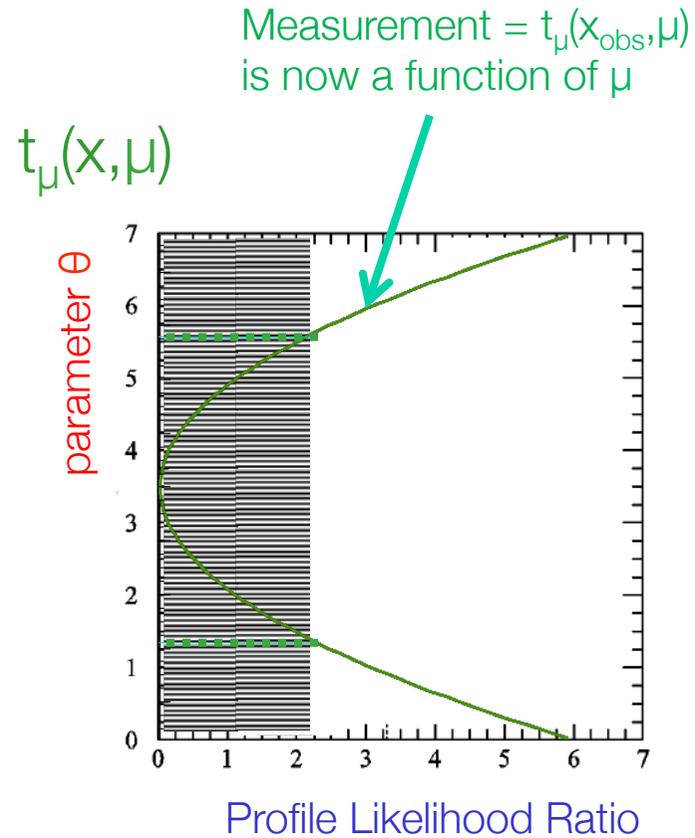
Profile scan of a Gaussian plus Polynomial probability model



PLR Confidence interval vs MINOS



Asymptotically,
distribution is identical
for all μ

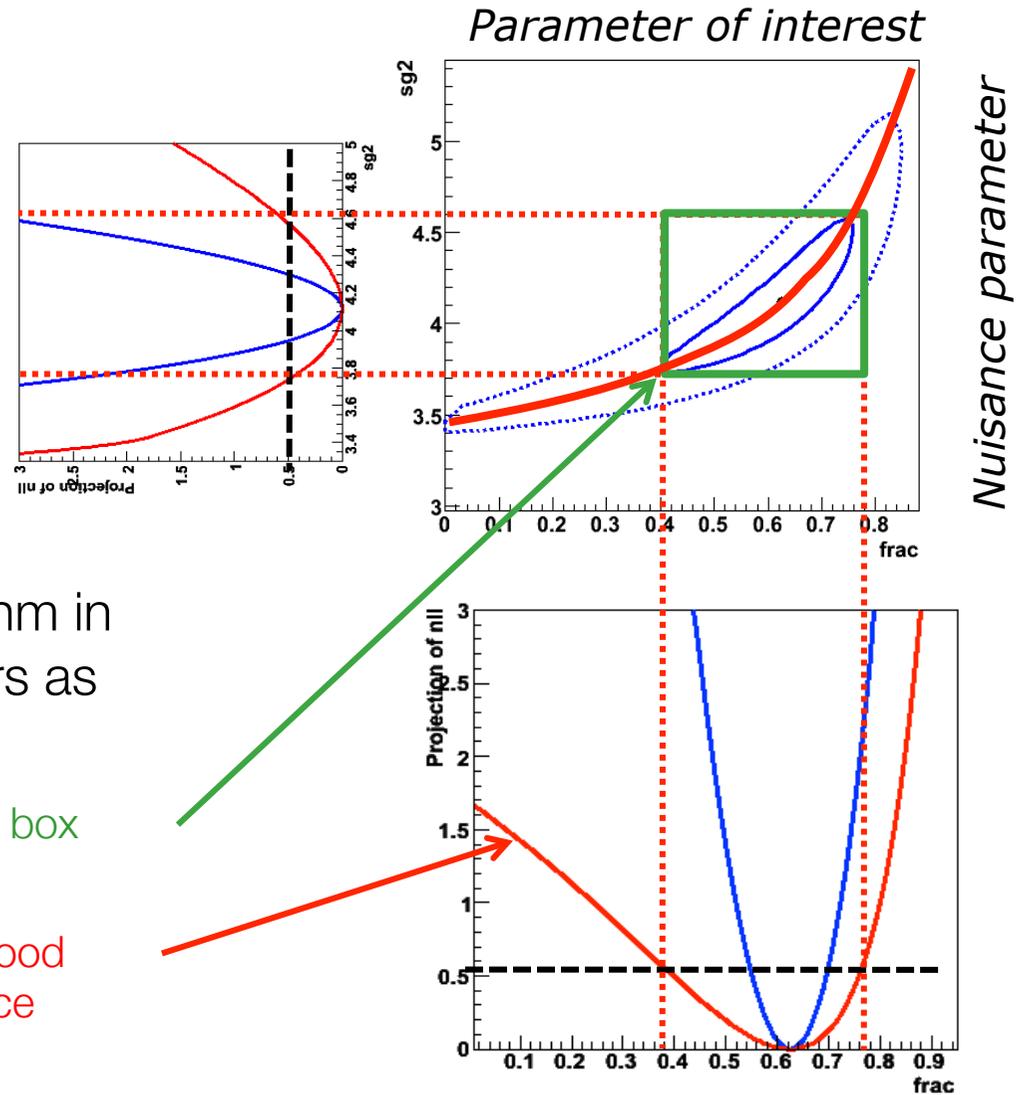


*NB: asymptotically, distribution
is also independent of true
values of θ*

$$f(t_\mu; \Lambda) = \frac{1}{2\sqrt{t_\mu}} \frac{1}{\sqrt{2\pi}} \left[\exp\left(-\frac{1}{2}(\sqrt{t_\mu} + \sqrt{\Lambda})^2\right) + \exp\left(-\frac{1}{2}(\sqrt{t_\mu} - \sqrt{\Lambda})^2\right) \right]$$

$$\Lambda = \frac{(\mu - \mu')^2}{\sigma^2}$$

Link between MINOS errors and profile likelihood



- Note that MINOS algorithm in MINUIT gives same errors as Profile Likelihood Ratio
 - MINOS errors is bounding box around $\lambda(s)$ contour
 - Profile Likelihood = Likelihood minimized w.r.t. all nuisance parameters

NB: Similar to graphical interpretation of variance estimators, but those always assume an elliptical contour from a perfectly parabolic likelihood

Summary on NPs in confidence intervals

- Exact confidence intervals are difficult with nuisance parameters
 - Interval should cover for any value of nuisance parameters
 - Technically difficult and significant over-coverage common
- LHC solution Profile Likelihood ratio → Guaranteed coverage at *measured* values of nuisance parameters only
 - Technically replace likelihood ratio with profile likelihood ratio
 - Computationally more intensive (need to minimize likelihood w.r.t all nuisance parameters for each evaluation of the test statistic), but still very tractable
- Asymptotically confidence intervals constructed with profile likelihood ratio test statistics correspond to (MINOS) likelihood ratio intervals
 - As distribution of profile likelihood becomes asymptotically independent of θ , coverage for all values of θ restored

Dealing with nuisance parameters in Bayesian intervals

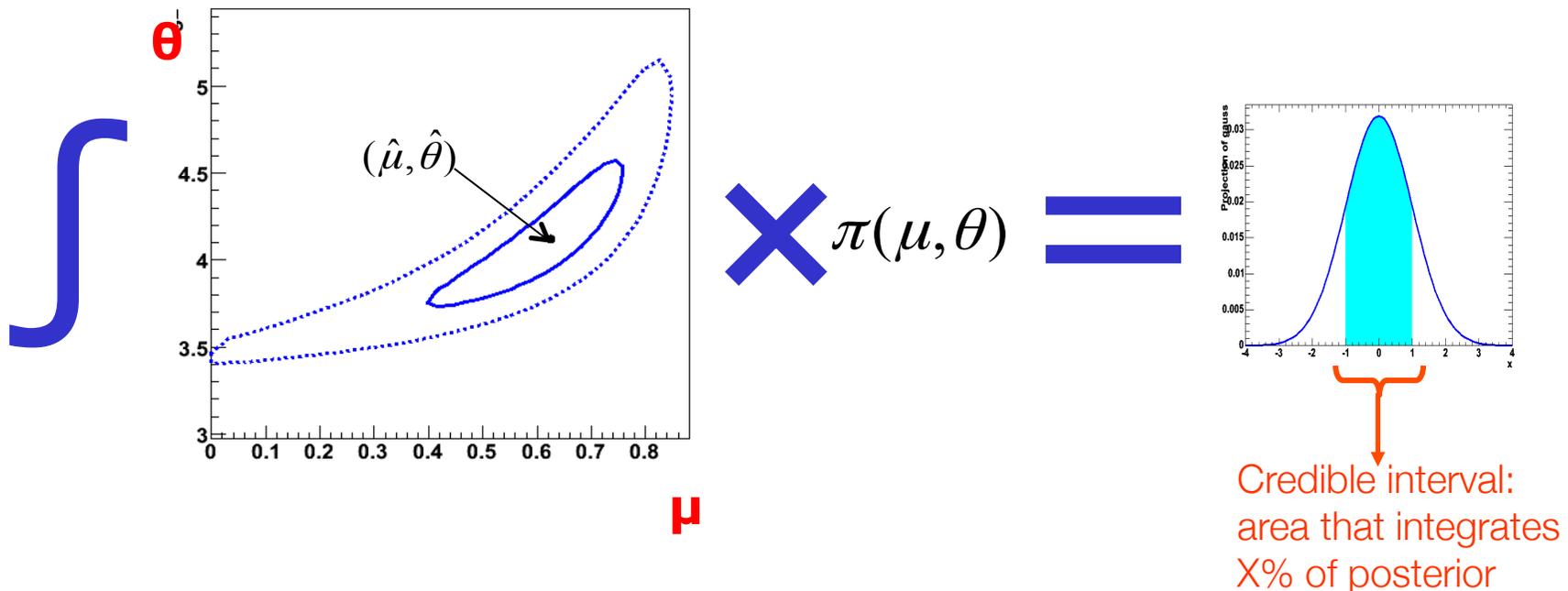
- Elimination of nuisance parameters in Bayesian interval: **Integrate over the full subspace of all nuisance parameters;**

$$P(\mu | x) \propto L(x | \mu) \cdot \pi(\mu)$$



$$P(\mu | x) \propto \int \left(L(x | \mu, \vec{\theta}) \pi(\mu) \pi(\vec{\theta}) \right) d\vec{\theta}$$

- You are left with posterior pdf for μ



Computational aspects of dealing with nuisance parameters

- Dealing with many nuisance parameters is computationally intensive in both Bayesian and (LHC) Frequentist approach
- Profile Likelihood approach
 - Computational challenge = **Minimization** of likelihood w.r.t. all nuisance parameters for every point in the profile likelihood curve
 - Minimization can be a difficult problem, e.g. if there are strong correlations, or multiple minima
- Bayesian approach
 - Computational challenge = **Integration** of posterior density of all nuisance parameters
 - Requires sampling of very potentially very large space.
 - Markov Chain MC and importance sampling techniques can help, but still very CPU consuming

Other procedures that have been tried*

- Hybrid Frequentist-Bayesian approach ('Cousins-Highland / Z_N ')
 - Integrate likelihood over nuisance parameters

$$L_m(\mu) = \int \left(L(\mu, \vec{\theta}) \pi(\vec{\theta}) \right) d\vec{\theta}$$

- Then treat integrated L_m as test statistic \rightarrow obtain p-value from its distribution
- In practice integral is performed using MC integration, so often described as a 'sampling method'

$$L_m(\mu) = \frac{1}{N} \sum_{MC} L(\mu, \vec{\theta}_i) \pi(\vec{\theta}_i)$$

- Method has been shown to have bad coverage

- Ad-hoc sampling methods of various types.

- Usually amount to either MC integration or fancy error propagation

Note that sampling the conditional estimator $\hat{\mu} \Big|_{\theta}$ over sample of θ values obtained from $\pi(\theta)$ is just glorified error propagation!

* But are known to have problems

How much do answers differ between methods?

A Prototype Problem

What is significance Z of an observation $x = 178$ events in a signal like region, if my expected background $b = 100$ with a 10% uncertainty?

- if you use the ATLAS TDR formula $Z_5 = 5.5$
- if you use Cousins-Highland $Z_N = 5.0$

The question seems simple enough, but it is not actually well-posed

- what do I mean by 10% background uncertainty?

Typically, we consider an auxiliary measurement y used to estimate background (Type I systematic)

- eg: a sideband counting experiment where background in sideband is a factor τ bigger than in signal region

$$L_P(x, y | \mu, b) = \text{Pois}(x | \mu + b) \cdot \text{Pois}(y | \tau b).$$

These slide discuss a 'prototype' likelihood that statisticians like:

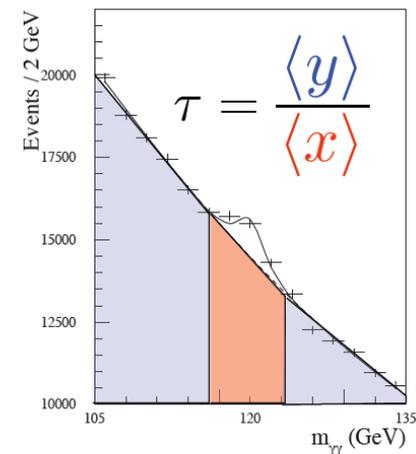
$$\text{Poisson}(N_{\text{sig}} | s+b) \cdot \text{Poisson}(N_{\text{ctl}} | \tau \cdot b)$$

NB: This is one of the very few problems with nuisance parameters with can be exactly calculation

Example Sideband Measurement

Sideband measurement used to extrapolate / interpolate the background rate in signal-like region

For now ignore uncertainty in extrapolation.



$$L_P(x, y | \mu, b) = \text{Pois}(x | \mu + b) \cdot \text{Pois}(y | \tau b).$$

Recent comparisons results from PhyStat 2007

Comparison of Methods for Prototype Problem

In my contribution to PhyStat2005, I considered this problem and compared the coverage for several methods

- ▶ See Linnemann's PhyStat03 paper

Major results:

- ▶ Cousins-Highland result (Z_N) badly under-covers (only 4.2σ)!
 - rate of Type I error is 110 times higher than stated!
 - much less luminosity required

▶ Profile Likelihood Ratio (MINUIT/MINOS) works great out to 5σ !

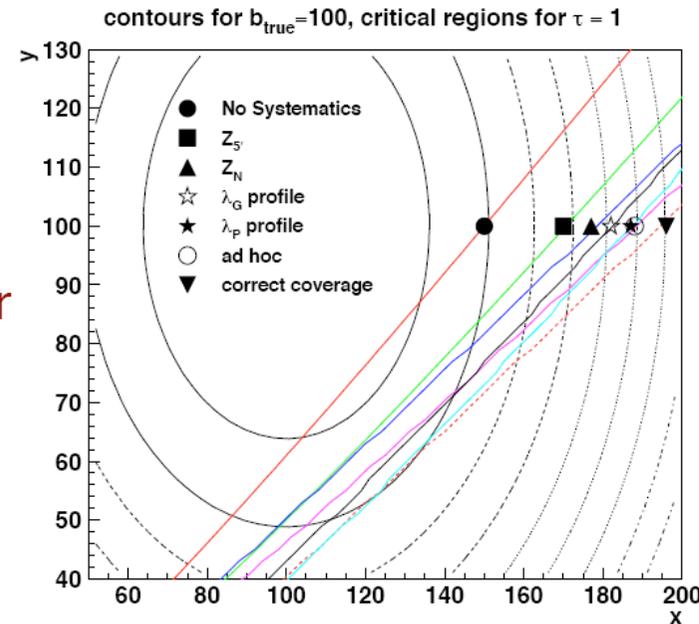


Figure 7. A comparison of the various methods critical boundary $x_{crit}(y)$ (see text). The concentric ovals represent contours of L_G from Eq. 15.

Method	$L_G (Z\sigma)$	$L_P (Z\sigma)$	$x_{crit}(y = 100)$
No Syst	3.0	3.1	150
$Z_{5'}$	4.1	4.1	171
Z_N (Sec. 4.1)	4.2	4.2	178
<i>ad hoc</i>	4.6	4.7	188
$Z_\Gamma = Z_{Bi}$	4.9	5.0	185
profile λ_P	5.0	5.0	185
profile λ_G	4.7	4.7	~182

Exact solution

Summary of statistical treatment of nuisance parameters

- Each statistical method has an associated technique to propagate the effect of uncertain NPs on the estimate of the POI
 - Parameter estimation → Joint unconditional estimation
 - Variance estimation → Replace d^2L/dp^2 with Hessian matrix
 - Hypothesis tests & confidence intervals → Use profile likelihood ratio
 - Bayesian credible intervals → Integration ('Marginalization')
- Be sure to use the right procedure with the right method
 - Anytime you integrate a Likelihood you are a Bayesian
 - If you are minimizing the likelihood you are usually a Frequentist
 - If you sample something chances are you performing either a (Bayesian) Monte Carlo integral, or are doing glorified error propagation
- Answers can differ substantially between methods!
 - This is not always a problem, but can also be a consequence of a difference in the problem statement